

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Akihiko OHWADA

Serial No.: To Be Assigned

Filed: May 31, 2000

For: PIPELINE OPERATOR

)
)
)
)
)
)
)

Group Art Unit: To Be Assigned

Examiner: To Be Assigned

10023 U.S. PTO
09/586961



#3

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN
APPLICATION IN ACCORDANCE
WITH THE REQUIREMENTS OF 37 C.F.R. §1.55**

*Assistant Commissioner for Patents
Washington, D.C. 20231*

Sir:

In accordance with the provisions of 37 C.F.R. § 1.55, Applicants submit herewith a certified copy of the following foreign application:

Japanese Patent Application No. 11-284850 filed October 5, 1999.

It is respectfully requested that Applicants be given the benefit of the foreign filing date, as evidenced by the certified papers attached hereto, in accordance with the requirements of 35 U.S.C. § 119.

Respectfully submitted,

STAAS & HALSEY, LLP

Dated: May 31, 2000

By: _____

James D. Halsey, Jr.
Registration No. 22,729

700 Eleventh Street, N.W., Suite 500
Washington, D.C. 20001
(202) 434-1500

日 本 国 特 許 庁
PATENT OFFICE
JAPANESE GOVERNMENT



別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日
Date of Application:

1999年10月 5日

出 願 番 号
Application Number:

平成11年特許願第284850号

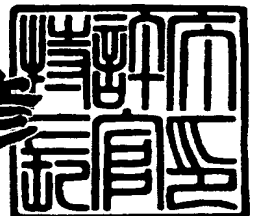
出 願 人
Applicant(s):

富士通株式会社

2000年 5月19日

特許庁長官
Commissioner,
Patent Office

近 藤 隆 彦



出証番号 出証特2000-3036619

【書類名】 特許願

【整理番号】 9901475

【提出日】 平成11年10月 5日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 9/38 310
G06F 9/38 370

【発明の名称】 パイプライン演算装置

【請求項の数】 10

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 大和田 昭彦

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100089118

【弁理士】

【氏名又は名称】 酒井 宏明

【手数料の表示】

【予納台帳番号】 036711

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9717671

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 パイプライン演算装置

【特許請求の範囲】

【請求項 1】 少なくとも第 1 演算ステージおよび第 2 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージおよび前記第 2 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 1 演算手段と、

前記第 2 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 2 演算手段と、

前記第 1 演算手段および前記第 2 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、前記第 2 演算手段に対して発行された前記命令をデコードする際に、前記第 1 演算手段に対して、上流のラッチ手段に保持される前記被演算データをスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 2】 少なくとも第 1 演算ステージおよび第 2 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージおよび前記第 2 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 1 演算手段と、

前記第 2 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 2 演算手段と、

前記第 1 演算手段および前記第 2 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、前記第 1 演算手段に対して発行された前記命令をデコードする際に、前記第 2 演算手段に対して、上流のラッチ手段に保持される前記第 1 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 3】 第 1 演算ステージ～第 n (> 1) 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～第 n 演算手段と、

前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、第 x ($1 < x$) 演算手段に対して発行された前記命令をデコードする際に、前記第 1 演算手段～第 $x - 1$ 演算手段に対して、上流のラッチ手段に保持される前記被演算データをスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 4】 第 1 演算ステージ～第 n (> 1) 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステー

ジ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～第 n 演算手段と、

前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、第 x ($x < n$) 演算手段に対して発行された前記命令をデコードする際に、第 x + 1 演算手段～前記第 n 演算手段に対して、上流のラッチ手段に保持される前記第 x 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 5】 少なくとも第 1 演算ステージおよび第 2 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージおよび前記第 2 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 1 演算手段と、

前記第 2 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 2 演算手段と、

前記第 1 演算手段および前記第 2 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、前記第 1 演算手段および前記第 2 演算手段に対して、それぞれの演算に関して前記第 1 演算手段の演算結果を前記第 2 演算手段の被

演算データとするような相関関係がある命令をデコードし、また、前記第 2 演算手段に対して単独で演算を行う命令をデコードする際に、前記第 1 演算手段に対して、上流のラッチ手段に保持される被演算データをスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 6】 少なくとも第 1 演算ステージおよび第 2 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージおよび前記第 2 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 1 演算手段と、

前記第 2 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 2 演算手段と、

前記第 1 演算手段および前記第 2 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、前記第 1 演算手段および前記第 2 演算手段に対して、それぞれの演算に関して前記第 1 演算手段の演算結果を前記第 2 演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 1 演算手段に対して単独で演算を行う命令をデコードする際に、前記第 2 演算手段に対して、上流のラッチ手段に保持される前記第 1 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 7】 第 1 演算ステージ～第 n (> 1) 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～第 n 演算手段と、

前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、前記第 1 演算手段～前記第 n 演算手段のうち、 m ($< n$) 段からなる第 r (> 1) 演算手段～第 s ($r < s < n$) 演算手段に対して、それぞれの演算に関して前段の演算手段の演算結果を次段の演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 r 演算手段～前記第 s 演算手段における第 x ($r \leq x \leq s$) 演算手段に対して単独で演算を行う命令をデコードする際に、前記第 1 演算手段～第 $x-1$ 演算手段に対して、上流のラッチ手段に保持される被演算データをスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 8】 第 1 演算ステージ～第 n (> 1) 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～前記第 n 演算手段と、

前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、前記第 1 演算手段～前記第 n 演算手段のうち、 m ($< n$) 段からなる第 r (> 1) 演算手段～第 s ($r < s < n$) 演算手段に対して

、それぞれの演算に関して前段の演算手段の演算結果を次段の演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 r 演算手段～前記第 s 演算手段における第 x ($r \leq x \leq s$) 演算手段～第 $x+p$ ($p \leq s-r$) 演算手段に対して p 段の演算手段により一つの演算が完了するような命令をデコードする際に、前記第 1 演算手段～第 $x-1$ 演算手段に対して、上流のラッチ手段に保持される被演算データをスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 9】 第 1 演算ステージ～第 n (>1) 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～第 n 演算手段と、

前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、前記第 1 演算手段～前記第 n 演算手段のうち、 m ($<n$) 段からなる第 r (>1) 演算手段～第 s ($r < s < n$) 演算手段に対して、それぞれの演算に関して前段の演算手段の演算結果を次段の演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 r 演算手段～第 s 演算手段における第 x ($r \leq x \leq s$) 演算手段に対して単独で演算を行う命令をデコードする際に、第 $x+1$ 演算手段～前記第 n 演算手段に対して、上流のラッチ手段に保持される、前記第 x 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【請求項 10】 第 1 演算ステージ～第 n (>1) 演算ステージを有するパイプライン演算装置であって、

前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段と、

前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～第 n 演算手段と、

前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段と、

を備え、

前記命令デコード手段は、前記第 1 演算手段～前記第 n 演算手段のうち、 m ($< n$) 段からなる第 r (> 1) 演算手段～第 s ($r < s < n$) 演算手段に対して、それぞれの演算に関して前段の演算手段の演算結果を次段の演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 r 演算手段～第 s 演算手段における第 $x - p$ ($r \leq x \leq s$, $p \leq s - r$) 演算手段～第 x 演算手段に対して p 段の演算手段を使用して一つの演算を行う命令をデコードする際に、第 $x + 1$ 演算手段～前記第 n 演算手段に対して、上流のラッチ手段に保持される、前記第 $x - p$ 演算手段～前記第 x 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とするパイプライン演算装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、パイプライン処理を用いたパイプライン演算装置に関するものであり、特に、ステージラッチ回路、ステージ間配線、制御回路等のハードウェア量、消費電力の低減が可能なパイプライン演算装置に関するものである。

【0002】

近時においては、より複雑な演算を高速で実行するためのマイクロプロセッサが求められており、このような高速化を実現するためのマイクロプロセッサとしては、パイプライン処理を用いたパイプライン演算装置が用いられる。このパイ

プライン処理は、複数の命令実行の開始時点を 1 クロックずつずらすことでオーバラップさせ、等価的に命令実行を高速化するための処理である。ここで、パイプライン演算装置をコンピュータシステムに実装する際には、実装面積、消費電力、価格等が検討の要素として重要な意味をもっており、小型、低消費電力、低価格という条件を満たすパイプライン演算装置が要請されている。

【0003】

【従来の技術】

図 27 は、従来のパイプライン演算装置の構成を示す図である。この図に示したパイプライン演算装置は、第 1 演算ステージおよび第 2 演算ステージという 2 段の演算ステージを有している。これら第 1 演算ステージおよび第 2 演算ステージには、後述する演算器 22_1 および演算器 12_2 がそれぞれ設けられている。被演算データ用レジスタ 10 は、パイプラインパス P_{10} に対応して設けられており、第 1 被演算データ（ソース・オペランド）を保持する。被演算データ用レジスタ 20 は、パイプラインパス P_{20} に対応して設けられており、第 2 被演算データ（ソース・オペランド）を保持する。命令用レジスタ 30 は、命令パイプラインパス P_{30} に対応して設けられており、命令 $INST_1$ 、命令 $INST_2$ を保持する。命令 $INST_1$ は、演算器 22_1 に対して発行される演算命令であり、命令 $INST_2$ は、演算器 12_2 に対して発行される演算命令である。

【0004】

また、従来のパイプライン演算装置は、2 段の演算ステージの入力段、第 1 演算ステージと第 2 演算ステージとの間（演算ステージ間）、および 2 段の演算ステージの出力段には、データ、命令等を一時的に保持する複数のステージラッチ回路がそれぞれ設けられている。具体的には、入力段には、ステージラッチ回路 11_1 、ステージラッチ回路 21_1 およびステージラッチ回路 31_1 が並列的にそれぞれ設けられており、演算ステージ間には、ステージラッチ回路 11_2 、ステージラッチ回路 21_2 およびステージラッチ回路 31_2 が並列的にそれぞれ設けられている。出力段には、ステージラッチ回路 11_3 が設けられている。また、これらのステージラッチ回路 11_1 、ステージラッチ回路 11_2 等は、入力段→演算ステージ間→出力段という具合に、1 クロックサイクルをおいて順次駆動

される。

【0 0 0 5】

第1演算ステージにおいて、ステージラッチ回路 $1\ 1_1$ は、被演算データ用レジスタ $1\ 0$ に対応して設けられており、被演算データ用レジスタ $1\ 0$ からの第1被演算データを保持する。ステージラッチ回路 $2\ 1_1$ は、被演算データ用レジスタ $2\ 0$ に対応して設けられており、被演算データ用レジスタ $2\ 0$ からの第2被演算データを保持する。ステージラッチ回路 $3\ 1_1$ は、命令用レジスタ $3\ 0$ に対応して設けられており、命令用レジスタ $3\ 0$ からの命令 $INST_1$ 、命令 $INST_2$ を保持する。命令デコーダ $3\ 2_1$ は、ステージラッチ回路 $3\ 1_1$ からの命令 $INST_1$ をデコードする。演算器 $2\ 2_1$ は、命令 $INST_1$ がデコードされたとき、ステージラッチ回路 $1\ 1_1$ の値（第1被演算データ）およびステージラッチ回路 $2\ 1_1$ の値（第2被演算データ）を用いて、命令 $INST_1$ に対応する演算を行い演算結果をパイプラインパス P_{20} を介してステージラッチ回路 $2\ 1_2$ へ出力する。

【0 0 0 6】

第2演算ステージにおいて、ステージラッチ回路 $1\ 1_2$ は、被演算データ用レジスタ $1\ 0$ に対応して設けられており、ステージラッチ回路 $1\ 1_1$ の値（第1被演算データ）を保持する。ステージラッチ回路 $2\ 1_2$ は、被演算データ用レジスタ $2\ 0$ に対応して設けられており、演算器 $2\ 2_1$ の演算結果を保持する。ステージラッチ回路 $3\ 1_2$ は、命令用レジスタ $3\ 0$ に対応して設けられており、ステージラッチ回路 $3\ 1_1$ の値（命令 $INST_1$ 、命令 $INST_2$ ）を保持する。命令デコーダ $3\ 2_2$ は、ステージラッチ回路 $3\ 1_2$ からの命令 $INST_2$ をデコードする。演算器 $1\ 2_2$ は、命令 $INST_2$ がデコードされたとき、ステージラッチ回路 $1\ 1_2$ の値（第1被演算データ）を用いて、命令 $INST_2$ に対応する演算を行い演算結果をパイプラインパス P_{10} を介して出力する。

【0 0 0 7】

マルチプレクサ $4\ 0_2$ は、命令デコーダ $3\ 2_2$ により切り替え制御される2入力1出力型の切替器であり、演算器 $1\ 2_2$ の演算結果、ステージラッチ回路 $2\ 1_2$ の値（演算器 $2\ 2_1$ の演算結果）のうちいずれか一方を出力する。具体的には

、マルチプレクサ 40_2 は、命令デコーダ 32_2 により命令 $INST_2$ がデコードされたとき演算器 12_2 の演算結果を出力する。一方、マルチプレクサ 40_2 は、命令 $INST_2$ 以外の命令（たとえば、命令 $INST_1$ ）が命令デコーダ 32_2 に入力されたとき、ステージラッチ回路 21_2 の値を出力する。ステージラッチ回路 11_3 は、マルチプレクサ 40_2 の切り替え状態に応じて、演算器 12_2 の演算結果またはステージラッチ回路 21_2 の値（演算器 22_1 の演算結果）を保持する。演算結果用レジスタ 50 は、ステージラッチ回路 11_3 の値、すなわち、パイプライン演算装置における演算結果（デスティネーション・オペランド）を保持する。

【0008】

つぎに上述した従来のパイプライン演算装置の動作について説明する。はじめに、命令 $INST_1$ が発行された場合の動作について説明する。最初のクロックにおいて、被演算データ用レジスタ 10 、被演算データ用レジスタ 20 および命令用レジスタ 30 に、第1被演算データ、第2被演算データおよび命令 $INST_1$ がそれぞれ保持されると、つぎのクロックでは、上記第1被演算データ、第2被演算データおよび命令 $INST_1$ がステージラッチ回路 11_1 、ステージラッチ回路 21_1 およびステージラッチ回路 31_1 にそれぞれ保持される。これにより、命令デコーダ 32_1 は、ステージラッチ回路 31_1 からの命令 $INST_1$ をデコードする。演算器 22_1 は、ステージラッチ回路 11_1 の値（第1被演算データ）と、ステージラッチ回路 21_1 の値（第2被演算データ）とを用いて、命令 $INST_1$ に応じた演算を行う。

【0009】

そして、つぎのクロックでは、ステージラッチ回路 11_1 の値（第1被演算データ）、演算器 22_1 の演算結果およびステージラッチ回路 31_1 の値（命令 $INST_1$ ）が、ステージラッチ回路 11_2 、ステージラッチ回路 21_2 およびステージラッチ回路 31_2 にそれぞれ保持される。このとき、命令デコーダ 32_2 は、ステージラッチ回路 31_2 の値（命令 $INST_1$ ）が、演算器 12_2 に応じた命令と無関係であるため、マルチプレクサ 40_2 にステージラッチ回路 21_2 を選択させる。

【0 0 1 0】

そして、つぎのクロックでは、ステージラッチ回路 $1\ 1_3$ には、マルチプレクサ $4\ 0_2$ を介して、ステージラッチ回路 $2\ 1_2$ の値（演算器 $2\ 2_1$ の演算結果）が保持される。これにより、演算結果用レジスタ $5\ 0$ には、パイプライン演算装置の演算結果として、演算器 $2\ 2_1$ の演算結果が保持される。

【0 0 1 1】

つぎに、命令 $INST_2$ が発行された場合の動作について説明する。最初のクロックにおいて、被演算データ用レジスタ $1\ 0$ 、被演算データ用レジスタ $2\ 0$ および命令用レジスタ $3\ 0$ に、第 1 被演算データ、第 2 被演算データおよび命令 $INST_2$ がそれぞれ保持されると、つぎのクロックでは、上述した動作と同様に、上記第 1 被演算データ、第 2 被演算データおよび命令 $INST_2$ がステージラッチ回路 $1\ 1_1$ 、ステージラッチ回路 $2\ 1_1$ およびステージラッチ回路 $3\ 1_1$ にそれぞれ保持される。このとき、命令デコーダ $3\ 2_1$ は、ステージラッチ回路 $3\ 1_1$ の値（命令 $INST_2$ ）が演算器 $2\ 2_1$ に無関係な命令であるため、デコードを行わない。したがって、この場合には、演算器 $2\ 2_1$ は、いずれの演算も行わない。

【0 0 1 2】

そして、つぎのクロックでは、ステージラッチ回路 $1\ 1_1$ の値（第 1 被演算データ）、ステージラッチ回路 $3\ 1_1$ の値（命令 $INST_2$ ）が、ステージラッチ回路 $1\ 1_2$ 、およびステージラッチ回路 $3\ 1_2$ にそれぞれ保持される。このとき、命令デコーダ $3\ 2_2$ は、ステージラッチ回路 $3\ 1_2$ からの命令 $INST_2$ をデコードするとともに、マルチプレクサ $4\ 0_2$ において演算器 $1\ 2_2$ 側を選択させる。これにより、演算器 $1\ 2_2$ は、ステージラッチ回路 $1\ 1_2$ の値（第 1 被演算データ）を用いて、命令 $INST_2$ に応じた演算を行う。

【0 0 1 3】

そして、つぎのクロックでは、ステージラッチ回路 $1\ 1_3$ には、マルチプレクサ $4\ 0_2$ を介して、演算器 $1\ 2_2$ の演算結果が保持される。これにより、演算結果用レジスタ $5\ 0$ には、パイプライン演算装置の演算結果として、演算器 $1\ 2_2$ の演算結果が保持される。

【0 0 1 4】

【発明が解決しようとする課題】

ところで、前述したように、従来のパイプライン演算装置においては、図 2 7 に示した被演算データ用レジスタ 1 0 からの第 1 被演算データ（ソース・オペランド）を保持するステージラッチ回路 1 1₂ が設けられているとともに、演算器 2 2₁ の演算結果を保持するステージラッチ回路 2 1₂ が上記ステージラッチ回路 1 1₂ とは独立して設けられている。したがって、このような構成では、被演算データ用レジスタ 1 0 に保持された第 1 被演算データは、ステージラッチ回路 1 1₁ およびステージラッチ回路 1 1₂ をスルーして演算器 1 2₂ に入力される。一方、演算器 2 2₁ の演算結果は、ステージラッチ回路 2 1₂ をスルーしてマルチプレクサ 4 0₂ に入力される。

【0 0 1 5】

このように、従来のパイプライン演算装置においては、第 1 演算ステージと第 2 演算ステージとの間に設けられているステージラッチ回路 1 1₂ およびステージラッチ回路 2 1₂ がそれぞれ独立しているため、おのずと回路構成（ステージラッチ回路、配線、制御回路）が冗長になってしまう。したがって、従来のパイプライン演算装置においては、冗長構成が採られていることにより、ハードウェア量、消費電力が大きいという問題があった。

【0 0 1 6】

本発明は、上記に鑑みてなされたもので、ハードウェア量、消費電力を低減することができるパイプライン演算装置を提供することを目的とする。

【0 0 1 7】

【課題を解決するための手段】

上記目的を達成するために、請求項 1 にかかる発明は、少なくとも第 1 演算ステージおよび第 2 演算ステージを有するパイプライン演算装置であって、前記第 1 演算ステージおよび前記第 2 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態 1 のステージラッチ回路 1 1 0₁、ステージラッチ回路 1 1 0₂ およびステージラッチ回路 1 1 0₃ に相当）と、前記第

1 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 1 演算手段（後述する実施の形態 1 の演算器 1 2 0₁ に相当）と、前記第 2 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 2 演算手段（後述する実施の形態 1 の演算器 1 2 0₂ に相当）と、前記第 1 演算手段および前記第 2 演算手段に対して発行された前記命令をデコードする命令デコード手段（後述する実施の形態 1 の命令デコーダ 3 2 0₁ および命令デコーダ 3 2 0₂ に相当）とを備え、前記命令デコード手段は、前記第 2 演算手段に対して発行された前記命令をデコードする際に、前記第 1 演算手段に対して、上流のラッチ手段に保持される前記被演算データをスルーさせる命令として前記命令をデコードすることを特徴とする。

【 0 0 1 8 】

この請求項 1 にかかる発明によれば、命令デコード手段により、第 2 演算手段に対して命令をデコードする際に、第 1 演算手段に対して被演算データをスルーさせる命令がデコードされると、第 1 演算手段は、上流のラッチ手段に保持されている被演算データをスルーで下流のラッチ手段へ出力する。これにより、被演算データは、上記下流のラッチ手段に保持される。そして、第 2 演算手段は、上流の上記ラッチ手段に保持された被演算データを用いて、命令に応じた演算を行う。

【 0 0 1 9 】

このように、請求項 1 にかかる発明によれば、第 2 演算手段に対して命令をデコードする際に、第 1 演算手段において、上流のラッチ手段に保持される被演算データをスルーさせるようにしたので、第 1 演算手段と第 2 演算手段との間で、ラッチ手段の共有、および配線量の低減が可能となることから、ハードウェア量および消費電力が低減する。

【 0 0 2 0 】

また、請求項 2 にかかる発明は、少なくとも第 1 演算ステージおよび第 2 演算ステージを有するパイプライン演算装置であって、前記第 1 演算ステージおよび

前記第 2 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態 2 のステージラッチ回路 110_1 、ステージラッチ回路 110_2 よびステージラッチ回路 110_3 に相当）と、前記第 1 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 1 演算手段（後述する実施の形態 2 の演算器 120_1 に相当）と、前記第 2 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 2 演算手段（後述する実施の形態 2 の演算器 120_2 に相当）と、前記第 1 演算手段および前記第 2 演算手段に対して発行された前記命令をデコードする命令デコード手段（後述する実施の形態 2 の命令デコーダ 320_1 および命令デコーダ 320_2 に相当）とを備え、前記命令デコード手段は、前記第 1 演算手段に対して発行された前記命令をデコードする際に、前記第 2 演算手段に対して、上流のラッチ手段に保持される前記第 1 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とする。

【0021】

この請求項 2 にかかる発明によれば、命令デコード手段により、第 1 演算手段に対して命令をデコードする際に、第 2 演算手段に対して、第 1 演算手段の演算結果をスルーさせる命令がデコードされると、第 1 演算手段は、上流のラッチ手段に保持されている被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する。これにより、第 1 演算手段の演算結果は、上記下流のラッチ手段に保持される。そして、第 2 演算手段は、上流の上記ラッチ手段に保持された第 1 演算手段の演算結果をスルーさせる。

【0022】

このように、請求項 2 にかかる発明によれば、第 1 演算手段に対して命令をデコードする際に、第 2 演算手段において、上流のラッチ手段に保持される第 1 演算手段の演算結果をスルーさせるようにしたので、第 2 演算手段の下流におけるラッチ手段および配線量の低減が可能となることから、ハードウェア量および消

費電力が低減する。

【0023】

また、請求項3にかかる発明は、第1演算ステージ～第 n (>1) 演算ステージを有するパイプライン演算装置であって、前記第1演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態3のステージラッチ回路 110_1 ～ステージラッチ回路 110_{n+1} に相当）と、前記第1演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第1演算手段～第 n 演算手段（後述する実施の形態3の演算器 120_1 ～演算器 120_n に相当）と、前記第1演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段（後述する実施の形態3の命令デコーダ 320_{11} ～命令デコーダ 320_n に相当）とを備え、前記命令デコード手段は、第 x ($1 < x$) 演算手段に対して発行された前記命令をデコードする際に、前記第1演算手段～第 $x-1$ 演算手段に対して、上流のラッチ手段に保持される前記被演算データをスルーさせる命令として前記命令をデコードすることを特徴とする。

【0024】

この請求項3にかかる発明によれば、命令デコード手段により、第 x 演算手段に対して命令をデコードする際に、第1演算手段～第 $x-1$ 演算手段に対して被演算データをスルーさせる命令がデコードされると、第1演算手段～第 $x-1$ 演算手段は、上流のラッチ手段に保持されている被演算データをスルーで下流のラッチ手段へ順次出力する。これにより、被演算データは、第1演算手段～第 $x-1$ 演算手段におけるそれぞれのラッチ手段に順次保持される。そして、第 x 演算手段は、上流の上記ラッチ手段に保持された被演算データを用いて、命令に応じた演算を行う。

【0025】

このように、請求項3にかかる発明によれば、第 x 演算手段に対して命令をデコードする際に、第1演算手段～第 $x-1$ 演算手段において、上流のラッチ手段

に保持される被演算データをスルーさせるようにしたので、第 1 演算ステージ～第 $x - 1$ 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する。

【0026】

また、請求項 4 にかかる発明は、第 1 演算ステージ～第 n (> 1) 演算ステージを有するパイプライン演算装置であって、前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態 4 のステージラッチ回路 110₁～ステージラッチ回路 110_{n+1} に相当）と、前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～第 n 演算手段（後述する実施の形態 4 の演算器 120₁～演算器 120_n に相当）と、前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段（後述する実施の形態 4 の命令デコーダ 320₁₁～命令デコーダ 320_n に相当）とを備え、前記命令デコード手段は、第 x ($x < n$) 演算手段に対して発行された前記命令をデコードする際に、第 $x + 1$ 演算手段～前記第 n 演算手段に対して、上流のラッチ手段に保持される前記第 x 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とする。

【0027】

この請求項 4 にかかる発明によれば、命令デコード手段により、第 x 演算手段に対して命令をデコードする際に、第 $x + 1$ 演算手段～第 n 演算手段に対して、第 x 演算手段の演算結果をスルーさせる命令がデコードされると、第 x 演算手段は、上流のラッチ手段に保持されている被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する。これにより、第 x 演算手段の演算結果は、上記下流のラッチ手段に保持される。そして、第 $x + 1$ 演算手段～第 n 演算手段は、上流の上記ラッチ手段に保持された第 x 演算手段の演算結果を順次スルーさせる。

【0028】

このように、請求項 4 にかかる発明によれば、第 x 演算手段に対して命令をデコードする際に、第 $x + 1$ 演算手段～第 n 演算手段において、上流のラッチ手段に保持される第 x 演算手段の演算結果をスルーさせるようにしたので、第 $x + 1$ 演算ステージ～第 n 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する。

【0029】

また、請求項 5 にかかる発明は、少なくとも第 1 演算ステージおよび第 2 演算ステージを有するパイプライン演算装置であって、前記第 1 演算ステージおよび前記第 2 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態 5 のステージラッチ回路 110_1 、ステージラッチ回路 110_2 およびステージラッチ回路 110_3 に相当）と、前記第 1 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 1 演算手段（後述する実施の形態 5 の演算器 120_1 に相当）と、前記第 2 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 2 演算手段（後述する実施の形態 5 の演算器 120_2 に相当）と、前記第 1 演算手段および前記第 2 演算手段に対して発行された前記命令をデコードする命令デコード手段（後述する実施の形態 5 の命令デコーダ 320_1 および命令デコーダ 320_2 に相当）とを備え、前記命令デコード手段は、前記第 1 演算手段および前記第 2 演算手段に対して、それぞれの演算に関して前記第 1 演算手段の演算結果を前記第 2 演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 2 演算手段に対して単独で演算を行う命令をデコードする際に、前記第 1 演算手段に対して、上流のラッチ手段に保持される被演算データをスルーさせる命令として前記命令をデコードすることを特徴とする。

【0030】

この請求項 5 にかかる発明によれば、命令デコード手段により、第 1 演算手段および第 2 演算手段に対して相関関係がある命令がデコードされると、第 1 演算

手段では、上記命令に応じた演算が行われ、この演算結果は、下流のラッチ手段に保持される。つぎに、第 2 演算手段では、上記ラッチ手段に保持された演算結果を用いて、上記命令に応じた演算が行われ、この演算結果は、下流のラッチ手段に保持される。また、請求項 5 にかかる発明によれば、第 2 演算手段に対して単独で演算を行わせる命令をデコードする際に、第 1 演算手段に対して被演算データをスルーさせる命令がデコードされると、第 1 演算手段は、上流のラッチ手段に保持されている被演算データをスルーで下流のラッチ手段へ出力する。これにより、被演算データは、上記下流のラッチ手段に保持される。そして、第 2 演算手段は、上流の上記ラッチ手段に保持された被演算データを用いて、命令に応じた演算を行う。

【 0 0 3 1 】

このように、請求項 5 にかかる発明によれば、第 2 演算手段に対して命令をデコードする際に、第 1 演算手段において、上流のラッチ手段に保持される被演算データをスルーさせるようにしたので、第 1 演算手段と第 2 演算手段との間で、ラッチ手段の共有、および配線量の低減が可能となることから、ハードウェア量および消費電力が低減する。また、請求項 5 にかかる発明によれば、第 2 演算手段において、第 1 演算手段とは独立して演算を行うことができる。

【 0 0 3 2 】

また、請求項 6 にかかる発明は、少なくとも第 1 演算ステージおよび第 2 演算ステージを有するパイプライン演算装置であって、前記第 1 演算ステージおよび前記第 2 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態 6 のステージラッチ回路 1 1 0₁、ステージラッチ回路 1 1 0₂ およびステージラッチ回路 1 1 0₃ に相当）と、前記第 1 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 1 演算手段（後述する実施の形態 6 の演算器 1 2 0₁ に相当）と、前記第 2 演算ステージに設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する第 2 演算手段（後述する実

施の形態 6 の演算器 120_2 に相当) と、前記第 1 演算手段および前記第 2 演算手段に対して発行された前記命令をデコードする命令デコード手段(後述する実施の形態 6 の命令デコーダ 320_1 および命令デコーダ 320_2 に相当) とを備え、前記命令デコード手段は、前記第 1 演算手段および前記第 2 演算手段に対して、それぞれの演算に関して前記第 1 演算手段の演算結果を前記第 2 演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 1 演算手段に対して単独で演算を行う命令をデコードする際に、前記第 2 演算手段に対して、上流のラッチ手段に保持される前記第 1 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とする。

【0033】

この請求項 6 にかかる発明によれば、命令デコード手段により、第 1 演算手段および第 2 演算手段に対して相関関係がある命令がデコードされると、第 1 演算手段では、上記命令に応じた演算が行われ、この演算結果は、下流のラッチ手段に保持される。つぎに、第 2 演算手段では、上記ラッチ手段に保持された演算結果を用いて、上記命令に応じた演算が行われ、この演算結果は、下流のラッチ手段に保持される。また、請求項 6 にかかる発明によれば、第 1 演算手段に対して単独で演算を行わせる命令をデコードする際に、第 2 演算手段に対して第 1 演算手段の演算結果をスルーさせる命令がデコードされると、第 1 演算手段は、上流のラッチ手段に保持されている被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する。これにより、第 1 演算手段の演算結果は、上記下流のラッチ手段に保持される。そして、第 2 演算手段は、上流の上記ラッチ手段に保持された第 1 演算手段の演算結果をスルーさせる。

【0034】

このように、請求項 6 にかかる発明によれば、第 1 演算手段に対して命令をデコードする際に、第 2 演算手段において、上流のラッチ手段に保持される第 1 演算手段の演算結果をスルーさせるようにしたので、第 2 演算手段の下流におけるラッチ手段および配線量の低減が可能となることから、ハードウェア量および消費電力が低減する。また、請求項 6 にかかる発明によれば、第 1 演算手段において、第 2 演算手段とは独立して演算を行うことができる。

【0035】

また、請求項7にかかる発明は、第1演算ステージ～第 n (>1) 演算ステージを有するパイプライン演算装置であって、前記第1演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態7のステージラッチ回路 110_1 ～ステージラッチ回路 110_{n+1} に相当）と、前記第1演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第1演算手段～第 n 演算手段（後述する実施の形態7の演算器 120_1 ～演算器 120_n に相当）と、前記第1演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段（後述する実施の形態7の命令デコーダ 320_{11} ～命令デコーダ 320_n に相当）とを備え、前記命令デコード手段は、前記第1演算手段～前記第 n 演算手段のうち、 m ($<n$) 段からなる第 r (>1) 演算手段～第 s ($r < s < n$) 演算手段に対して、それぞれの演算に関して前段の演算手段の演算結果を次段の演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 r 演算手段～前記第 s 演算手段における第 x ($r \leq x \leq s$) 演算手段に対して単独で演算を行う命令をデコードする際に、前記第1演算手段～第 $x-1$ 演算手段に対して、上流のラッチ手段に保持される被演算データをスルーさせる命令として前記命令をデコードすることを特徴とする。

【0036】

この請求項7にかかる発明によれば、命令デコード手段により、 m 段からなる第 r 演算手段～第 s 演算手段に対して相関関係がある命令がデコードされると、第 r 演算手段～第 s 演算手段では、上記命令に応じた演算がそれぞれ行われ、演算結果は、下流のラッチ手段に順次保持される。また、請求項7にかかる発明によれば、第 x 演算手段に対して命令をデコードする際に、第1演算手段～第 $x-1$ 演算手段に対して被演算データをスルーさせる命令がデコードされると、第1演算手段～第 $x-1$ 演算手段は、上流のラッチ手段に保持されている被演算データをスルーで下流のラッチ手段へ順次出力する。これにより、被演算データは、

第 1 演算手段～第 $x-1$ 演算手段におけるそれぞれのラッチ手段に順次保持される。そして、第 x 演算手段は、上流の上記ラッチ手段に保持された被演算データを用いて、命令に応じた演算を行う。

【0037】

このように、請求項 7 にかかる発明によれば、第 x 演算手段に対して命令をデコードする際に、第 1 演算手段～第 $x-1$ 演算手段において、上流のラッチ手段に保持される被演算データをスルーさせるようにしたので、第 1 演算ステージ～第 $x-1$ 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する。また、請求項 7 にかかる発明によれば、第 r 演算手段～第 s 演算手段における第 x 演算手段の演算を、他の演算手段とは独立させて行うことができる。

【0038】

また、請求項 8 にかかる発明は、第 1 演算ステージ～第 n (> 1) 演算ステージを有するパイプライン演算装置であって、前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態 7 のステージラッチ回路 110_1 ～ステージラッチ回路 110_{n+1} に相当）と、前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～前記第 n 演算手段（後述する実施の形態 7 の演算器 120_1 ～演算器 120_n に相当）と、前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段（後述する実施の形態 7 の命令デコーダ 320_{11} ～命令デコーダ 320_n に相当）とを備え、前記命令デコード手段は、前記第 1 演算手段～前記第 n 演算手段のうち、 m ($< n$) 段からなる第 r (> 1) 演算手段～第 s ($r < s < n$) 演算手段に対して、それぞれの演算に関して前段の演算手段の演算結果を次段の演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 r 演算手段～前記第 s 演算手段における第 x ($r \leq x \leq s$) 演算手段～第 $x+p$ ($p \leq s-r$) 演算手段に対して p 段の演算手段により一

つの演算が完了するような命令をデコードする際に、前記第 1 演算手段～第 $x - 1$ 演算手段に対して、上流のラッチ手段に保持される被演算データをスルーさせる命令として前記命令をデコードすることを特徴とする。

【0039】

この請求項 8 にかかる発明によれば、命令デコード手段により、 m 段からなる第 r 演算手段～第 s 演算手段に対して相関関係がある命令がデコードされると、第 r 演算手段～第 s 演算手段では、上記命令に応じた演算がそれぞれ行われ、演算結果は、下流のラッチ手段に順次保持される。また、請求項 8 にかかる発明によれば、第 x 演算手段～第 $x + p$ 演算手段に対して命令をデコードする際に、第 1 演算手段～第 $x - 1$ 演算手段に対して被演算データをスルーさせる命令がデコードされると、第 1 演算手段～第 $x - 1$ 演算手段は、上流のラッチ手段に保持されている被演算データをスルーで下流のラッチ手段へ順次出力する。これにより、被演算データは、第 1 演算手段～第 $x - 1$ 演算手段におけるそれぞれのラッチ手段に順次保持される。そして、第 x 演算手段～第 $x + p$ 演算手段は、上流の上記ラッチ手段に保持された被演算データを用いて、命令に応じた演算を行う。

【0040】

このように、請求項 8 にかかる発明によれば、第 x 演算手段～第 $x + p$ 演算手段に対して命令をデコードする際に、第 1 演算手段～第 $x - 1$ 演算手段において、上流のラッチ手段に保持される被演算データをスルーさせるようにしたので、第 1 演算ステージ～第 $x - 1$ 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する。また、請求項 8 にかかる発明によれば、第 r 演算手段～第 s 演算手段における第 x 演算手段～第 $x + p$ 演算手段の演算を、他の演算手段とは独立させて行うことができる。

【0041】

また、請求項 9 にかかる発明は、第 1 演算ステージ～第 n (> 1) 演算ステージを有するパイプライン演算装置であって、前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施

の形態 8 のステージラッチ回路 $1\ 1\ 0_1 \sim$ ステージラッチ回路 $1\ 1\ 0_{n+1}$ に相当) と、前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～第 n 演算手段(後述する実施の形態 8 の演算器 $1\ 2\ 0_1 \sim$ 演算器 $1\ 2\ 0_n$ に相当) と、前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段(後述する実施の形態 8 の命令デコーダ $3\ 2\ 0_{11} \sim$ 命令デコーダ $3\ 2\ 0_n$ に相当) とを備え、前記命令デコード手段は、前記第 1 演算手段～前記第 n 演算手段のうち、 $m (< n)$ 段からなる第 $r (> 1)$ 演算手段～第 $s (r < s < n)$ 演算手段に対して、それぞれの演算に関して前段の演算手段の演算結果を次段の演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 r 演算手段～第 s 演算手段における第 $x (r \leq x \leq s)$ 演算手段に対して単独で演算を行う命令をデコードする際に、第 $x + 1$ 演算手段～前記第 n 演算手段に対して、上流のラッチ手段に保持される、前記第 x 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とする。

【0042】

この請求項 9 にかかる発明によれば、命令デコード手段により、 m 段からなる第 r 演算手段～第 s 演算手段に対して相関関係がある命令がデコードされると、第 r 演算手段～第 s 演算手段では、上記命令に応じた演算がそれぞれ行われ、演算結果は、下流のラッチ手段に順次保持される。また、請求項 9 にかかる発明によれば、命令デコード手段により、第 x 演算手段に対して命令をデコードする際に、第 $x + 1$ 演算手段～第 n 演算手段に対して、第 x 演算手段の演算結果をスルーさせる命令がデコードされると、第 x 演算手段は、上流のラッチ手段に保持されている被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する。これにより、第 x 演算手段の演算結果は、上記下流のラッチ手段に保持される。そして、第 $x + 1$ 演算手段～第 n 演算手段は、上流の上記ラッチ手段に保持された第 x 演算手段の演算結果を順次スルーさせる。

【0043】

このように、請求項 9 にかかる発明によれば、第 x 演算手段に対して命令をデ

コードする際に、第 $x+1$ 演算手段～第 n 演算手段において、上流のラッチ手段に保持される第 x 演算手段の演算結果をスルーさせるようにしたので、第 $x+1$ 演算ステージ～第 n 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する。また、請求項 9 にかかる発明によれば、第 r 演算手段～第 s 演算手段における第 x 演算手段の演算を、他の演算手段とは独立させて行うことができる。

【0044】

また、請求項 10 にかかる発明は、第 1 演算ステージ～第 n (>1) 演算ステージを有するパイプライン演算装置であって、前記第 1 演算ステージ～前記第 n 演算ステージにおける、入力段、演算ステージ間、出力段にそれぞれ設けられ、被演算データまたは演算結果をそれぞれ保持する複数のラッチ手段（後述する実施の形態 8 のステージラッチ回路 110₁～ステージラッチ回路 110_{n+1} に相当）と、前記第 1 演算ステージ～前記第 n 演算ステージにそれぞれ設けられ、上流の前記ラッチ手段に保持された前記被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へそれぞれ出力する第 1 演算手段～第 n 演算手段（後述する実施の形態 8 の演算器 120₁～演算器 120_n に相当）と、前記第 1 演算手段～前記第 n 演算手段に対して発行された前記命令をデコードする命令デコード手段（後述する実施の形態 8 の命令デコーダ 320₁₁～命令デコーダ 320_n に相当）とを備え、前記命令デコード手段は、前記第 1 演算手段～前記第 n 演算手段のうち、 m ($<n$) 段からなる第 r (>1) 演算手段～第 s ($r < s < n$) 演算手段に対して、それぞれの演算に関して前段の演算手段の演算結果を次段の演算手段の被演算データとするような相関関係がある命令をデコードし、また、前記第 r 演算手段～第 s 演算手段における第 $x-p$ ($r \leq x \leq s$ 、 $p \leq s-r$) 演算手段～第 x 演算手段に対して p 段の演算手段を使用して一つの演算を行う命令をデコードする際に、第 $x+1$ 演算手段～前記第 n 演算手段に対して、上流のラッチ手段に保持される、前記第 $x-p$ 演算手段～前記第 x 演算手段の演算結果をスルーさせる命令として前記命令をデコードすることを特徴とする。

【0045】

この請求項 1 0 にかかる発明によれば、命令デコード手段により、 m 段からなる第 r 演算手段～第 s 演算手段に対して相関関係がある命令がデコードされると、第 r 演算手段～第 s 演算手段では、上記命令に応じた演算がそれぞれ行われ、演算結果は、下流のラッチ手段に順次保持される。また、請求項 1 0 にかかる発明によれば、命令デコード手段により、第 $x - p$ 演算手段～第 x 演算手段に対して命令をデコードする際に、第 $x + 1$ 演算手段～第 n 演算手段に対して、第 $x - p$ 演算手段～第 x 演算手段の演算結果をスルーさせる命令がデコードされると、第 $x - p$ 演算手段～第 x 演算手段は、上流のラッチ手段に保持されている被演算データを用いて、命令に応じた演算を行い、演算結果を下流のラッチ手段へ出力する。これにより、第 $x - p$ 演算手段～第 x 演算手段の演算結果は、上記下流のラッチ手段に保持される。そして、第 $x + 1$ 演算手段～第 n 演算手段は、上流の上記ラッチ手段に保持された第 $x - p$ 演算手段～第 x 演算手段の演算結果を順次スルーさせる。

【0 0 4 6】

このように請求項 1 0 にかかる発明によれば、第 $x - p$ 演算手段～第 x 演算手段に対して命令をデコードする際に、第 $x + 1$ 演算手段～第 n 演算手段において、上流のラッチ手段に保持される第 $x - p$ 演算手段～第 x 演算手段の演算結果をスルーさせるようにしたので、第 $x + 1$ 演算ステージ～第 n 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する。また、請求項 1 0 にかかる発明によれば、第 r 演算手段～第 s 演算手段における第 $x - p$ 演算手段～第 x 演算手段の演算を、他の演算手段とは独立させて行うことができる。

【0 0 4 7】

【発明の実施の形態】

以下、図面を参照して本発明にかかるパイプライン演算装置の実施の形態 1 ～ 8 について詳細に説明する。

【0 0 4 8】

(実施の形態 1)

図 1 は、本発明にかかる実施の形態 1 の構成を示すブロック図である。この図

に示したパイプライン演算装置は、第 1 演算ステージおよび第 2 演算ステージという 2 段の演算ステージを有している。この第 1 演算ステージには、演算器 120_1 が設けられており、第 2 演算ステージには、演算器 120_2 が設けられている。被演算データ用レジスタ 100 は、第 1 被演算データ（ソース・オペランド） $SOURCE_1$ を保持する。被演算データ用レジスタ 200 は、第 2 被演算データ（ソース・オペランド）を保持する。命令用レジスタ 300 は、命令パイプラインパス P_{300} に対応して設けられており、命令 $INST_1$ 、命令 $INST_2$ を保持する。命令 $INST_1$ は、演算器 120_1 に対して発行される演算命令であり、命令 $INST_2$ は、演算器 120_2 に対して発行される演算命令である。

【0049】

また、実施の形態 1 のパイプライン演算装置は、2 段の演算ステージの入力段、第 1 演算ステージと第 2 演算ステージとの間（演算ステージ間）、および 2 段の演算ステージの出力段には、データ、命令等を一時的に保持するステージラッチ回路がそれぞれ設けられている。具体的には、入力段には、ステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 が並列的にそれぞれ設けられており、演算ステージ間には、ステージラッチ回路 110_2 、およびステージラッチ回路 310_2 が並列的にそれぞれ設けられている。出力段には、ステージラッチ回路 110_3 が設けられている。また、これらのステージラッチ回路 110_1 、ステージラッチ回路 110_2 等は、入力段→演算ステージ間→出力段という具合に、1 クロックサイクルをおいて順次駆動される。

【0050】

ここで、前述した従来のパイプライン演算装置の構成を示した図 27 と図 1 とにおいて演算ステージ間（第 1 演算ステージと第 2 演算ステージとの間）のステージラッチ回路の数を比較すると、図 27 では、ステージラッチ回路の数が 3 個（ステージラッチ回路 11_2 、ステージラッチ回路 21_2 およびステージラッチ回路 31_2 ）であるのに対して、図 1 では、ステージラッチ回路の数が 2 個（ステージラッチ回路 110_2 およびステージラッチ回路 310_2 ）である。

【0051】

第1演算ステージにおいて、ステージラッチ回路 110_1 は、被演算データ用レジスタ 100 （パイプラインパス P_{100} ）に対応して設けられており、被演算データ用レジスタ 100 からの第1被演算データ $SOURCE_1$ を保持する。ステージラッチ回路 210_1 は、被演算データ用レジスタ 200 に対応して設けられており、被演算データ用レジスタ 200 からの第2被演算データを保持する。ステージラッチ回路 310_1 は、命令用レジスタ 300 に対応して設けられており、命令用レジスタ 300 からの命令 $INST_1$ 、命令 $INST_2$ を保持する。命令デコーダ 320_1 は、ステージラッチ回路 310_1 からの命令 $INST_1$ をデコードする。演算器 120_1 は、命令 $INST_1$ がデコードされたとき、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_1 の値（第2被演算データ）を用いて、命令 $INST_1$ に対応する演算を行い演算結果をステージラッチ回路 110_2 へ出力する。

【0052】

また、命令デコーダ 320_1 は、命令 $INST_1$ 以外の命令（この場合、命令 $INST_2$ ）が入力された場合、これをスルー命令 SRC_1 としてデコードし、演算器 120_1 を制御する。演算器 120_1 は、上記スルー命令 SRC_1 が入力された場合、二つの被演算データ（第1被演算データ $SOURCE_1$ 、第2被演算データ）のうち、第1被演算データ $SOURCE_1$ のみをスルーさせる。

【0053】

第2演算ステージにおいて、ステージラッチ回路 110_2 は、被演算データ用レジスタ 100 に対応して設けられており、演算器 120_1 の演算結果、または演算器 120_1 の出力値（第1被演算データ $SOURCE_1$ ）を保持する。ステージラッチ回路 310_2 は、命令用レジスタ 300 に対応して設けられており、ステージラッチ回路 310_1 の値（命令 $INST_1$ 、命令 $INST_2$ ）を保持する。命令デコーダ 320_2 は、ステージラッチ回路 310_2 からの命令 $INST_2$ をデコードする。演算器 120_2 は、命令 $INST_2$ がデコードされたとき、ステージラッチ回路 110_2 の値を用いて、命令 $INST_2$ に対応する演算を行い演算結果を出力する。バイパスライン B_2 は、演算器 120_2 に対して並列的に設けられており、ステージラッチ回路 110_2 の値を、演算器 120_2 を経由

させることなく、スルーでマルチプレクサ 400_2 に導く役目をしている。

【0054】

マルチプレクサ 400_2 は、命令デコーダ 320_2 により切り替え制御される 2 入力 1 出力型の切替器であり、演算器 120_2 の演算結果、ステージラッチ回路 110_2 の値のうちいずれか一方を出力する。具体的には、マルチプレクサ 400_2 は、命令デコーダ 320_2 により命令 $INST_2$ がデコードされたとき演算器 120_2 の演算結果を出力する。一方、マルチプレクサ 400_2 は、命令 $INST_2$ 以外の命令（この場合、命令 $INST_1$ ）が命令デコーダ 320_2 に入力されたとき、ステージラッチ回路 110_2 の値を出力する。ステージラッチ回路 110_3 は、マルチプレクサ 400_2 の切り替え状態に応じて、演算器 120_2 の演算結果またはステージラッチ回路 110_2 の値を保持する。演算結果用レジスタ 500 は、ステージラッチ回路 110_3 の値、すなわち、パイプライン演算装置における演算結果（デスティネーション・オペランド）を保持する。

【0055】

また、実施の形態 1 において、第 1 演算ステージの演算器 120_1 と、第 2 演算ステージの演算器 120_2 とは、互いに相関関係がない命令（命令 $INST_1$ 、命令 $INST_2$ ）に基づいて演算をそれぞれ行う。つまり、相関関係がない場合には、演算器 120_1 の演算と、演算器 120_2 の演算とがそれぞれ単独で行われる。さらに、実施の形態 1 においては、第 1 演算ステージでスルー命令 $SRCC_1$ がデコードされることで、第 1 被演算データ $SOURCE_1$ がスルーされる。

【0056】

つぎに、上述した実施の形態 1 の動作について図 2 に示したフローチャートを参照しつつ説明する。はじめに、命令 $INST_1$ が発行された場合の動作について説明する。図 2 に示したステップ SA1 では、被演算データ用レジスタ 100 、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第 1 被演算データ $SOURCE_1$ 、第 2 被演算データおよび命令 $INST_1$ がそれぞれ保持される。これにより、上記第 1 被演算データ $SOURCE_1$ 、第 2 被演算データおよび命令 $INST_1$ がステージラッチ回路 110_1 、ステージラッチ回路 210

1 およびステージラッチ回路 3 1 0₁ にそれぞれ保持される。

【0057】

つぎのステップ SA 2 では、命令デコーダ 3 2 0₁ は、ステージラッチ回路 3 1 0₁ に保持された命令の種類を判断する。この場合、ステージラッチ回路 3 1 0₁ に命令 INST₁ が保持されているため、命令デコーダ 3 2 0₁ は、ステップ SA 3 へ進む。ステップ SA 3 では、命令デコーダ 3 2 0₁ は、ステージラッチ回路 3 1 0₁ からの命令 INST₁ をデコードする。これにより、演算器 1 2 0₁ はステージラッチ回路 1 1 0₁ の値（第 1 被演算データ SOURCE₁）およびステージラッチ回路 2 1 0₁ の値（第 2 被演算データ）を用いて、命令 INST₁ に応じた演算を行う。

【0058】

つぎのステップ SA 4 では、演算器 1 2 0₁ の演算結果がステージラッチ回路 1 1 0₂ に保持される。このとき、ステージラッチ回路 3 1 0₁ の値（命令 INST₁）がステージラッチ回路 3 1 0₂ に保持される。つぎのステップ SA 8 では、命令デコーダ 3 2 0₂ は、ステージラッチ回路 3 1 0₂ に保持された命令の種類を判断する。この場合、ステージラッチ回路 3 1 0₂ に命令 INST₁ が保持されているため、命令デコーダ 3 2 0₂ は、ステップ SA 1 1 へ進む。

【0059】

ステップ SA 1 1 では、命令デコーダ 3 2 0₂ は、マルチプレクサ 4 0 0₂ にステージラッチ回路 1 1 0₂ を選択させる。これにより、ステップ SA 1 2 では、演算器 1 2 0₂ が動作せず、ステップ SA 1 3 では、ステージラッチ回路 1 1 0₂ の値（演算器 1 2 0₁ の演算結果）が、バイパスライン B₂ およびマルチプレクサ 4 0 0₂ を介してステージラッチ回路 1 1 0₃ に保持される。つぎのステップ SA 1 4 では、演算結果用レジスタ 5 0 0 には、パイプライン演算装置の演算結果として演算器 1 2 0₁ の演算結果が保持される。

【0060】

つぎに、命令 INST₂ が発行された場合の動作について説明する。図 2 に示したステップ SA 1 では、被演算データ用レジスタ 1 0 0、被演算データ用レジスタ 2 0 0 および命令用レジスタ 3 0 0 に、第 1 被演算データ SOURCE₁、

第2被演算データおよび命令 $INST_2$ がそれぞれ保持される。これにより、上記第1被演算データ $SOURCE_1$ 、第2被演算データおよび命令 $INST_2$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 にそれぞれ保持される。

【0061】

つぎのステップSA2では、命令デコーダ 320_1 は、ステージラッチ回路 310_1 に保持された命令の種類を判断する。この場合、ステージラッチ回路 310_1 に命令 $INST_2$ が保持されているため、命令デコーダ 320_1 は、ステップSA5へ進む。ステップSA5では、命令デコーダ 320_1 は、ステージラッチ回路 310_1 からの命令 $INST_2$ をスルー命令 SRC_1 に変換した後、ステップSA6へ進む。ステップSA6では、命令デコーダ 320_1 は、スルー命令 SRC_1 をデコードする。これにより、演算器 120_1 は、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）をスルーで通過させる。

【0062】

つぎのステップSA7では、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）がステージラッチ回路 110_2 に保持される。このとき、ステージラッチ回路 310_1 の値（命令 $INST_2$ ）がステージラッチ回路 310_2 に保持される。つぎのステップSA8では、命令デコーダ 320_2 は、ステージラッチ回路 310_2 に保持された命令の種類を判断する。この場合、ステージラッチ回路 310_2 に命令 $INST_2$ が保持されているため、命令デコーダ 320_2 は、ステップSA9へ進む。

【0063】

ステップSA9では、命令デコーダ 320_2 は、マルチプレクサ 400_2 に演算器 120_2 を選択させる。また、命令デコーダ 320_2 は、ステージラッチ回路 310_2 からの命令 $INST_2$ をデコードする。これにより、演算器 120_2 は、ステージラッチ回路 110_2 の値（第1被演算データ $SOURCE_1$ ）を用いて、命令 $INST_2$ に応じた演算を行う。つぎのステップSA10では、演算器 120_2 の演算結果がステージラッチ回路 110_3 に保持される。つぎのステップSA14では、演算結果用レジスタ 500 には、パイプライン演算装置の演

算結果として演算器 120₂ の演算結果が保持される。

【0064】

以上説明したように、実施の形態 1 によれば、命令デコーダ 320₁ において命令 INST₁ 以外の命令をスルー命令 SRC₁ に変換し、演算器 120₁ で第 1 被演算データ SOURCE₁ をスルーさせるとともに、第 2 演算ステージにバイパスライン B₂ を設けたので、演算器 120₁ と演算器 120₂ との間でステージラッチ回路 110₂ の共有、および配線量の低減が可能となる。したがって、実施の形態 1 によれば、ステージラッチ回路 110₂ の共有化により、ハードウェア量および消費電力が低減する。

【0065】

(実施の形態 2)

図 3 は、本発明にかかる実施の形態 2 の構成を示すブロック図である。この図において、図 1 の各部にそれぞれ対応する部分には同一の符号を付ける。ただし、図 3 に示した命令デコーダ 320₁ および命令デコーダ 320₂ の機能は、後述するように、図 1 に示した命令デコーダ 320₁ および命令デコーダ 320₂ の機能とは異なる。図 3 においては、第 1 演算ステージにマルチプレクサ 400₁ およびバイパスライン B₁ が新たに設けられている。さらに、図 3 においては、第 2 演算ステージに、図 1 に示したマルチプレクサ 400₂ およびバイパスライン B₂ が設けられていない。

【0066】

図 3 において、第 1 演算ステージの命令デコーダ 320₁ は、ステージラッチ回路 310₁ からの命令 INST₁ をデコードする。演算器 120₁ は、命令 INST₁ がデコードされたとき、ステージラッチ回路 110₁ の値、およびステージラッチ回路 210₁ の値を用いて、命令 INST₁ に対応する演算を行い演算結果 RESULT₁ を出力する。バイパスライン B₁ は、演算器 120₁ に対して並列的に設けられており、ステージラッチ回路 110₁ の値を、演算器 120₁ を経由させることなく、スルーでマルチプレクサ 400₁ に導く役目をしている。

【0067】

マルチプレクサ400₁は、命令デコーダ320₁により切り替え制御される2入力1出力型の切替器であり、演算器120₁の演算結果RESULT₁、ステージラッチ回路110₁の値のうちいずれか一方を出力する。具体的には、マルチプレクサ400₁は、命令デコーダ320₁により命令INST₁がデコードされたとき演算器120₁の演算結果RESULT₁を出力する。一方、マルチプレクサ400₁は、命令INST₁以外の命令（たとえば、命令INST₂）が命令デコーダ320₁に入力されたとき、ステージラッチ回路110₁の値を出力する。

【0068】

命令デコーダ320₂は、ステージラッチ回路310₂からの命令INST₂をデコードし、演算器120₂を制御する。演算器120₂は、命令INST₂がデコードされたとき、ステージラッチ回路110₁の値（第1被演算データ）を用いて、命令INST₂に対応する演算を行い演算結果をステージラッチ回路110₃へ出力する。

【0069】

また、命令デコーダ320₂は、命令INST₂以外の命令（この場合、命令INST₁）が入力された場合、これをスルー命令SRC₁に変換し、このスルー命令SRC₁をデコードする。演算器120₂は、上記スルー命令SRC₁が入力された場合、ステージラッチ回路110₂の値（この場合、演算結果RESULT₁）をスルーさせる。ステージラッチ回路110₃は、マルチプレクサ400₁の切り替え状態に応じて、ステージラッチ回路110₂の値（演算結果RESULT₁）または演算器120₂の演算結果を保持する。演算結果用レジスタ500は、ステージラッチ回路110₃の値、すなわち、パイプライン演算装置における演算結果（デスティネーション・オペランド）を保持する。

【0070】

また、実施の形態2において、第1演算ステージの演算器120₁と、第2演算ステージの演算器120₂とは、実施の形態1と同様にして、互いに相関関係がない命令（命令INST₁、命令INST₂）に基づいて演算をそれぞれ行う。つまり、相関関係がない場合には、演算器120₁の演算と、演算器120₂

の演算とがそれぞれ単独で行われる。さらに、実施の形態 2 においては、第 2 演算ステージでスルー命令 SRC_1 がデコードされることで、第 1 演算ステージにおける演算器 120_1 の演算結果 $RESULT_1$ がスルーされる。

【0071】

つぎに、上述した実施の形態 2 の動作について図 4 に示したフローチャートを参照しつつ説明する。はじめに、命令 $INST_1$ が発行された場合の動作について説明する。図 4 に示したステップ SB 1 では、被演算データ用レジスタ 100 、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第 1 被演算データ、第 2 被演算データおよび命令 $INST_1$ がそれぞれ保持される。これにより、上記第 1 被演算データ、第 2 被演算データおよび命令 $INST_1$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 にそれぞれ保持される。

【0072】

つぎのステップ SB 2 では、命令デコーダ 320_1 は、ステージラッチ回路 310_1 に保持された命令の種類を判断する。この場合、ステージラッチ回路 310_1 に命令 $INST_1$ が保持されているため、命令デコーダ 320_1 は、ステップ SB 3 へ進む。ステップ SB 3 では、命令デコーダ 320_1 は、マルチプレクサ 400_1 に演算器 120_1 を選択させる。また、命令デコーダ 320_1 は、ステージラッチ回路 310_1 からの命令 $INST_1$ をデコードし、演算器 120_1 を制御する。これにより、演算器 120_1 は、ステージラッチ回路 110_1 の値（第 1 被演算データ）、およびステージラッチ回路 210_1 の値（第 2 被演算データ）を用いて、命令 $INST_1$ に応じた演算を行う。つぎのステップ SB 4 では、演算器 120_1 の演算結果 $RESULT_1$ がマルチプレクサ 400_1 を介して、ステージラッチ回路 110_2 に保持される。これに並行して、ステージラッチ回路 310_1 の値（命令 $INST_1$ ）がステージラッチ回路 310_2 に保持される。

【0073】

つぎのステップ SB 8 では、命令デコーダ 320_2 は、ステージラッチ回路 310_2 に保持された命令の種類を判断する。この場合、ステージラッチ回路 310_2

0₂ に命令 INST₂ 以外の命令、すなわち、命令 INST₁ が保持されているため、命令デコーダ 320₂ は、ステップ SB11 へ進む。ステップ SB11 では、命令デコーダ 320₂ は、ステージラッチ回路 310₂ の値（命令 INST₁）を解読し、命令 INST₁ をスルー命令 SRC₁ に変換した後、ステップ SB12 へ進む。

【0074】

ステップ SB12 では、命令デコーダ 320₂ は、スルー命令 SRC₁ をデコードする。これにより、演算器 120₂ は、ステージラッチ回路 110₂ の値（演算結果 RESULT₁）をスルーで通過させる。つぎのステップ SB13 では、ステージラッチ回路 110₂ の値（演算結果 RESULT₁）がステージラッチ回路 110₃ に保持される。つぎのステップ SB14 では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120₁ の演算結果 RESULT₁ が保持される。

【0075】

つぎに、命令 INST₂ が発行された場合の動作について説明する。図 4 に示したステップ SB1 では、被演算データ用レジスタ 100、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第 1 被演算データ、第 2 被演算データおよび命令 INST₂ がそれぞれ保持される。これにより、上記第 1 被演算データ、第 2 被演算データおよび命令 INST₂ がステージラッチ回路 110₁、ステージラッチ回路 210₁ およびステージラッチ回路 310₁ にそれぞれ保持される。

【0076】

つぎのステップ SB2 では、命令デコーダ 320₁ は、ステージラッチ回路 310₁ に命令 INST₂ が保持されているため、ステップ SB5 へ進む。ステップ SB5 では、命令デコーダ 320₁ は、マルチプレクサ 400₁ にステージラッチ回路 110₁ を選択させる。ステップ SB6 では、演算器 120₁ が動作せず、ステップ SB7 では、ステージラッチ回路 110₁ の値（第 1 被演算データ）が、バイパスライン B₁ およびマルチプレクサ 400₁ を介してステージラッチ回路 110₂ に保持される。

【0077】

つぎのステップSB8では、命令デコーダ 320_2 は、ステージラッチ回路 310_2 に命令 $INST_2$ が保持されているため、ステップSB9へ進む。ステップSB9では、命令デコーダ 320_2 は、ステージラッチ回路 310_2 のからの命令 $INST_2$ をデコードする。演算器 120_2 は、ステージラッチ回路 110_2 の値（第1被演算データ）を用いて、命令 $INST_2$ に応じた演算を実行する。つぎのステップSB10では、演算器 120_2 の演算結果がステージラッチ回路 110_3 に保持され、ステップSB14では、演算結果用レジスタ500には、パイプライン演算装置の演算結果として演算器 120_2 の演算結果が保持される。

【0078】

以上説明したように、実施の形態2によれば、第1演算ステージにマルチプレクサ 400_1 を設けて、ステージラッチ回路 110_1 の値または演算器 120_1 の演算結果のいずれかをステージラッチ回路 110_2 に保持させるとともに、命令デコーダ 320_2 において命令 $INST_2$ 以外の命令をスルー命令 SRC_1 に変換し、演算器 120_2 で演算結果 $RESULT_1$ をスルーさせるようにしたので、演算器 120_2 の下流のステージラッチ回路および配線量の低減が可能となる。したがって、実施の形態2によれば、ハードウェア量および消費電力が低減する。

【0079】

(実施の形態3)

図5は、本発明にかかる実施の形態3の構成を示すブロック図である。この図においては、図1の各部に対応する部分には同一の符号をそれぞれ付ける。この図に示したパイプライン演算装置は、第1演算ステージ～第n演算ステージというn段の演算ステージを有している。ただし、図5には、第1演算ステージ、第 $x-1$ ($1 < x < n$) 演算ステージ、第x演算ステージ、第 $x+1$ 演算ステージおよび第n演算ステージが図示されており、他の演算ステージの図示が省略されている。

【0080】

また、実施の形態3においては、 n 段の演算ステージの入力段、演算ステージ間、および n 段の演算ステージの出力段には、データ、命令等を一時的に保持する複数のステージラッチ回路がそれぞれ設けられている。具体的には、入力段には、ステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 が並列的にそれぞれ設けられており、出力段には、ステージラッチ回路 110_{n+1} が設けられている。なお、演算ステージ間のステージラッチ回路については後述する。

【0081】

図5において、被演算データ用レジスタ 100 は、第1被演算データ（ソース・オペランド） $SOURCE_1$ を保持する。被演算データ用レジスタ 200 は、第2被演算データ（ソース・オペランド） $SOURCE_2$ を保持する。命令用レジスタ 300 は、各演算ステージにおいて用いられる命令 $INST_{11}$ 、命令 $INST_{12}$ 、命令 $INST_{x-11}$ 、 \dots 、命令 $INST_n$ をそれぞれ保持する。

【0082】

第1演算ステージにおいて、ステージラッチ回路 110_1 は、被演算データ用レジスタ 100 からの第1被演算データ $SOURCE_1$ を保持し、ステージラッチ回路 210_1 は、被演算データ用レジスタ 200 からの第2被演算データ $SOURCE_2$ を保持する。ステージラッチ回路 310_1 は、命令用レジスタ 300 に対応して設けられており、命令用レジスタ 300 からの命令 $INST_{11}$ 、命令 $INST_{12}$ 等を保持する。命令デコーダ 320_{11} は、ステージラッチ回路 310_1 からの命令 $INST_{11}$ をデコードする。

【0083】

上記演算器 120_1 は、命令 $INST_{11}$ がデコードされたとき、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）を用いて、命令 $INST_{11}$ に対応する演算を行い演算結果をステージラッチ回路 110_2 へ出力する。また、命令デコーダ 320_{11} は、命令 $INST_{11}$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 をデコードする。演算器 120_1 は、上記スルー命令 SRC_1 が入力された場合、第1被演算データ $SOURCE_1$ をスルーさせる。

【0084】

命令デコーダ 320_{12} は、ステージラッチ回路 310_1 からの命令 $INST_{12}$ をデコードする。この演算器 220_1 は、命令 $INST_{12}$ がデコードされたとき、ステージラッチ回路 210_1 の値（第2被演算データ $SOURCE_2$ ）を用いて、命令 $INST_{12}$ に対応する演算を行い演算結果をステージラッチ回路 210_2 へ出力する。また、命令デコーダ 320_{12} は、命令 $INST_{12}$ 以外の命令が入力された場合、これをスルー命令 SRC_2 に変換し、このスルー命令 SRC_2 をデコードする。演算器 220_1 は、上記スルー命令 SRC_2 が入力された場合、第2被演算データ $SOURCE_2$ をスルーさせる。

【0085】

ステージラッチ回路 110_2 、ステージラッチ回路 210_2 およびステージラッチ回路 310_2 は、第1演算ステージと第2演算ステージ（図示略）との間にそれぞれ設けられており、演算器 120_1 の出力値、演算器 220_1 の出力値およびステージラッチ回路 310_1 の値をそれぞれ保持する。以下、第2演算ステージ～第 $x-1$ 演算ステージまでは、上述した第1演算ステージと同一構成とされている。

【0086】

すなわち、第 $x-1$ 演算ステージにおいて、ステージラッチ回路 110_{x-1} は、第1被演算データ $SOURCE_1$ または第 $x-2$ 演算ステージ（図示略）の演算器の演算結果を保持する。ステージラッチ回路 210_{x-1} は、第2被演算データ $SOURCE_2$ または第 $x-2$ 演算ステージ（図示略）の演算器の演算結果を保持する。ステージラッチ回路 310_{x-1} は、第 $x-2$ 演算ステージにおけるステージラッチ回路の値（命令 $INST_{11}$ 、命令 $INST_{12}$ 等）を保持する。命令デコーダ 320_{x-11} は、ステージラッチ回路 310_{x-1} からの命令 $INST_{x-11}$ をデコードする。

【0087】

上記演算器 120_{x-1} は、命令 $INST_{x-11}$ がデコードされたとき、ステージラッチ回路 110_{x-1} の値を用いて、命令 $INST_{x-11}$ に対応する演算を行い演算結果をステージラッチ回路 110_x へ出力する。また、命令デコーダ 320_{x-}

11 は、命令 $INST_{x-11}$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_{x-1} に対してデコードする。演算器 120_{x-1} は、上記スルー命令 SRC_1 が入力された場合、ステージラッチ回路 110_{x-1} の値をスルーさせる。

【0088】

命令デコーダ 320_{x-12} は、ステージラッチ回路 310_{x-1} からの命令 $INST_{x-12}$ をデコードする。この演算器 220_{x-1} は、命令 $INST_{x-12}$ がデコードされたとき、ステージラッチ回路 210_{x-1} の値を用いて、命令 $INST_{x-12}$ に対応する演算を行い演算結果をステージラッチ回路 210_x へ出力する。また、命令デコーダ 320_{x-12} は、命令 $INST_{x-12}$ 以外の命令が入力された場合、これをスルー命令 SRC_2 に変換し、このスルー命令 SRC_2 を演算器 220_{x-1} に対してデコードする。演算器 220_{x-1} は、上記スルー命令 SRC_2 が入力された場合、ステージラッチ回路 210_{x-1} の値をスルーさせる。

【0089】

第 x 演算ステージにおいて、ステージラッチ回路 110_x は、第 1 被演算データ $SOURCE_1$ または第 $x-1$ 演算ステージにおける演算器 120_{x-1} の演算結果を保持する。ステージラッチ回路 210_x は、第 2 被演算データ $SOURCE_2$ または第 $x-1$ 演算ステージにおける演算器 220_{x-1} の演算結果を保持する。ステージラッチ回路 310_x は、ステージラッチ回路 310_{x-1} の値（命令 $INST_{11}$ 、命令 $INST_{12}$ 等）を保持する。命令デコーダ 320_x は、ステージラッチ回路 310_x からの命令 $INST_x$ をデコードする。

【0090】

上記演算器 120_x は、命令 $INST_x$ がデコードされたとき、ステージラッチ回路 110_x の値、およびステージラッチ回路 210_x の値を用いて、命令 $INST_x$ に対応する演算を行い演算結果をステージラッチ回路 110_{x+1} へ出力する。また、命令デコーダ 320_x は、命令 $INST_x$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_x に対してデコードする。演算器 120_x は、ステージラッチ回路 110_x の値、ステージラッチ回路 210_x の値のうち、ステージラッチ回路 110_x の

値のみをスルーさせる。

【0 0 9 1】

ここで、第 $x + 1$ 演算ステージ～第 n 演算ステージのそれぞれの構成は、前述した図 1 に示した第 2 演算ステージの構成と同様である。すなわち、第 $x + 1$ 演算ステージにおいて、ステージラッチ回路 $1 1 0_{x+1}$ は、第 1 被演算データ $S O U R C E_1$ または第 x 演算ステージにおける演算器 $1 2 0_x$ の演算結果を保持する。ステージラッチ回路 $3 1 0_{x+1}$ は、ステージラッチ回路 $3 1 0_x$ の値（命令 $I N S T_{11}$ 、命令 $I N S T_{12}$ 等）を保持する。命令デコーダ $3 2 0_{x+1}$ は、ステージラッチ回路 $3 1 0_{x+1}$ からの命令 $I N S T_{x+1}$ をデコードする。演算器 $1 2 0_{x+1}$ は、命令 $I N S T_{x+1}$ がデコードされたとき、ステージラッチ回路 $1 1 0_{x+1}$ の値を用いて、命令 $I N S T_{x+1}$ に対応する演算を行い演算結果を出力する。バイパスライン B_{x+1} は、演算器 $1 2 0_{x+1}$ に対して並列的に設けられており、ステージラッチ回路 $1 1 0_{x+1}$ の値を、演算器 $1 2 0_{x+1}$ を経由させることなく、スルーでマルチプレクサ $4 0 0_{x+1}$ に導く役目をしている。

【0 0 9 2】

マルチプレクサ $4 0 0_{x+1}$ は、命令デコーダ $3 2 0_{x+1}$ により切り替え制御される 2 入力 1 出力型の切替器であり、演算器 $1 2 0_{x+1}$ の演算結果、ステージラッチ回路 $1 1 0_{x+1}$ の値のうちいずれか一方を出力する。具体的には、マルチプレクサ $4 0 0_{x+1}$ は、命令デコーダ $3 2 0_{x+1}$ により命令 $I N S T_{x+1}$ がデコードされたとき演算器 $1 2 0_{x+1}$ の演算結果を出力する。一方、マルチプレクサ $4 0 0_{x+1}$ は、命令 $I N S T_{x+1}$ 以外の命令が命令デコーダ $3 2 0_{x+1}$ に入力されたとき、ステージラッチ回路 $1 1 0_{x+1}$ の値を出力する。ステージラッチ回路 $1 1 0_{x+2}$ は、マルチプレクサ $4 0 0_{x+1}$ の切り替え状態に応じて、演算器 $1 2 0_{x+1}$ の演算結果またはステージラッチ回路 $1 1 0_{x+1}$ の値を保持する。ステージラッチ回路 $1 1 0_{x+2}$ 、およびステージラッチ回路 $3 1 0_{x+2}$ は、第 $x + 1$ 演算ステージと第 $x + 2$ 演算ステージ（図示略）との間にそれぞれ設けられており、マルチプレクサ $4 0 0_{x+1}$ の出力値、およびステージラッチ回路 $3 1 0_{x+1}$ の値をそれぞれ保持する。以下、第 $x + 2$ 演算ステージ～第 n 演算ステージまでは、上述した第 $x + 1$ 演算ステージと同一構成とされている。

【0093】

すなわち、第 n 演算ステージにおいて、ステージラッチ回路 110_n は、第1被演算データ $SOURCE_1$ または第 $n-1$ 演算ステージ（図示略）における演算器の演算結果を保持する。ステージラッチ回路 310_n は、第 $n-1$ 演算ステージにおけるステージラッチ回路の値を保持する。命令デコーダ 320_n は、ステージラッチ回路 310_n からの命令 $INST_n$ をデコードする。演算器 120_n は、命令 $INST_n$ がデコードされたとき、ステージラッチ回路 110_n の値を用いて、命令 $INST_n$ に対応する演算を行い演算結果を出力する。バイパスライン B_n は、演算器 120_n に対して並列的に設けられており、ステージラッチ回路 110_n の値を、演算器 120_n を経由させることなく、スルーでマルチプレクサ 400_n に導く役目をしている。

【0094】

マルチプレクサ 400_n は、命令デコーダ 320_n により切り替え制御され、命令デコーダ 320_n により命令 $INST_n$ がデコードされたとき演算器 120_n の演算結果を出力する。一方、マルチプレクサ 400_n は、命令 $INST_n$ 以外の命令が命令デコーダ 320_n に入力されたとき、ステージラッチ回路 110_n の値を出力する。ステージラッチ回路 110_{n+1} は、マルチプレクサ 400_n の切り替え状態に応じて、演算器 120_n の演算結果またはステージラッチ回路 110_n の値を保持する。演算結果用レジスタ 500 は、ステージラッチ回路 110_{n+1} の値、すなわち、パイプライン演算装置における演算結果（デスティネーション・オペランド）を保持する。

【0095】

また、実施の形態3において、第1演算ステージ～第 n 演算ステージまでのそれぞれの演算器は、互いに相関関係がない命令に基づいて演算をそれぞれ行う。つまり、相関関係がない場合には、それぞれの演算器における演算が単独で行われる。さらに、実施の形態3においては、第1演算ステージ～第 $x-1$ 演算ステージでスルー命令 SRC_1 およびスルー命令 SRC_2 がデコードされることで、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ がスルーされる。

【0096】

つぎに、上述した実施の形態3の動作について図6～図8に示したフローチャートを参照しつつ説明する。はじめに、第 x 演算ステージの演算器 120_x に対して命令 $INST_x$ が発行された場合の動作について説明する。図6に示したステップSC1では、被演算データ用レジスタ100、被演算データ用レジスタ200および命令用レジスタ300に、第1被演算データ $SOURCE_1$ 、第2被演算データ $SOURCE_2$ および命令 $INST_x$ がそれぞれ保持される。これにより、上記第1被演算データ $SOURCE_1$ 、第2被演算データ $SOURCE_2$ および命令 $INST_x$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 に保持される。

【0097】

つぎのステップSC2では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステージラッチ回路 310_1 に保持された命令の種類を判断する。この場合、ステージラッチ回路 310_1 に命令 $INST_x$ が保持されているため、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステップSC5へ進む。なお、ステージラッチ回路 310_1 に命令 $INST_{11}$ および命令 $INST_{12}$ が保持されている場合、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステップSC3へ進み、命令 $INST_{11}$ および命令 $INST_{12}$ を演算器 120_1 および演算器 220_1 に対してそれぞれデコードする。これにより、演算器 120_1 および演算器 220_1 はステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_1 の値（第2被演算データ $SOURCE_2$ ）を用いて、命令 $INST_{11}$ および命令 $INST_{12}$ に応じた演算をそれぞれ行う。これにより、ステップSC4では、演算器 120_1 の演算結果、および演算器 220_1 の演算結果がステージラッチ回路 110_2 およびステージラッチ回路 210_2 に保持される。

【0098】

この場合、ステップSC5では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステージラッチ回路 310_1 の値（命令 $INST_x$ ）をそれぞれ解読し、命令 $INST_x$ をスルー命令 SRC_1 およびスルー命令 SRC_2 にそれぞれ

変換した後、ステップSC6へ進む。ステップSC6では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、演算器 120_1 および演算器 220_1 に対してスルー命令 SRC_1 およびスルー命令 SRC_2 をそれぞれデコードする。これにより、演算器 120_1 および演算器 220_1 は、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_1 の値（第2被演算データ $SOURCE_2$ ）をスルーでそれぞれ通過させる。

【0099】

つぎのステップSC7では、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_1 の値（第2被演算データ $SOURCE_2$ ）がステージラッチ回路 110_2 およびステージラッチ回路 210_2 にそれぞれ保持される。このとき、ステージラッチ回路 310_1 の値（命令 $INST_x$ ）がステージラッチ回路 310_2 に保持される。以後、第2演算ステージ～第 $x-2$ 演算ステージ（いずれも図示略）においては、第1演算ステージの場合と同様にして、スルー命令 SRC_1 およびスルー命令 SRC_2 がそれぞれデコードされることで、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ がそれぞれの演算ステージをスルーで順次通過する。また、第2演算ステージ～第 $x-2$ 演算ステージにおいては、命令 $INST_x$ がステージラッチ回路に順次保持される。

【0100】

そして、第 $x-1$ 演算ステージにおいて、ステップSC8では、命令デコーダ 320_{x-11} および命令デコーダ 320_{x-12} は、ステージラッチ回路 310_{x-1} に命令 $INST_x$ が保持されているため、ステップSC11へ進む。なお、ステージラッチ回路 310_{x-1} に命令 $INST_{x-11}$ および命令 $INST_{x-12}$ が保持されている場合、命令デコーダ 320_{x-11} および命令デコーダ 320_{x-12} は、ステップSC9へ進み、命令 $INST_{x-11}$ および命令 $INST_{x-12}$ を演算器 120_{x-1} および演算器 220_{x-1} に対してそれぞれデコードする。これにより、演算器 120_{x-1} および演算器 220_{x-1} は、ステージラッチ回路 110_{x-1} の値、およびステージラッチ回路 210_{x-1} の値を用いて、命令 $INST_{x-11}$ および命令 $INST_{x-12}$ に応じた演算をそれぞれ行う。そして、ステップSC10では、演

算器 120_{x-1} および演算器 220_{x-1} の演算結果がステージラッチ回路 110_x およびステージラッチ回路 210_x に保持される。

【0101】

この場合、ステップ SC11 では、命令デコーダ 320_{x-11} および命令デコーダ 320_{x-12} は、ステージラッチ回路 310_{x-1} の値（命令 $INST_x$ ）をそれぞれ解読し、命令 $INST_x$ をスルー命令 SRC_1 およびスルー命令 SRC_2 にそれぞれ変換した後、ステップ SC12 へ進む。ステップ SC12 では、命令デコーダ 320_{x-11} および命令デコーダ 320_{x-12} は、演算器 120_{x-1} および演算器 220_{x-1} に対してスルー命令 SRC_1 およびスルー命令 SRC_2 をそれぞれデコードする。これにより、演算器 120_{x-1} および演算器 220_{x-1} は、ステージラッチ回路 110_{x-1} の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_{x-1} の値（第2被演算データ $SOURCE_2$ ）をスルーでそれぞれ通過させる。

【0102】

つぎのステップ SC13 では、ステージラッチ回路 110_{x-1} の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_{x-1} の値（第2被演算データ $SOURCE_2$ ）がステージラッチ回路 110_x およびステージラッチ回路 210_x にそれぞれ保持される。このとき、ステージラッチ回路 310_{x-1} の値（命令 $INST_x$ ）がステージラッチ回路 310_x に保持される。

【0103】

つぎの第 x 演算ステージにおいて、ステップ SC14（図7参照）では、命令デコーダ 320_x は、ステージラッチ回路 310_x に命令 $INST_x$ が保持されているため、ステップ SC15 へ進む。なお、ステージラッチ回路 310_x に命令 $INST_x$ 以外の命令が保持されている場合、命令デコーダ 320_x は、ステップ SC17 へ進み、 $INST_x$ 以外の命令をスルー命令 SRC_1 に変換した後、ステップ SC18 へ進む。ステップ SC18 では、命令デコーダ 320_x は、演算器 120_x に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_x は、ステージラッチ回路 110_x の値をスルーで通過させる。つぎのステップ SC19 では、ステージラッチ回路 110_x の値がステージラッチ回路

110_{x+1} に保持される。

【0104】

この場合、ステップSC15では、命令デコーダ 320_x は、ステージラッチ回路 310_x からの命令 $INST_x$ を演算器 120_x に対してデコードする。これにより、演算器 120_x はステージラッチ回路 110_x の値（第1被演算データ $SOURCE_1$ ）、およびステージラッチ回路 210_x の値（第2被演算データ $SOURCE_2$ ）を用いて、命令 $INST_x$ に応じた演算を行う。これにより、ステップSC16では、演算器 120_x の演算結果がステージラッチ回路 110_{x+1} に保持される。このとき、ステージラッチ回路 310_x の値（命令 $INST_x$ ）がステージラッチ回路 310_{x+1} に保持される。

【0105】

つぎの第 $x+1$ 演算ステージにおいて、ステップSC20では、命令デコーダ 320_{x+1} は、ステージラッチ回路 310_{x+1} に命令 $INST_x$ が保持されているため、ステップSC23へ進む。なお、ステージラッチ回路 310_{x+1} に命令 $INST_{x+1}$ が保持されている場合、命令デコーダ 320_{x+1} は、ステップSC21へ進み、マルチプレクサ 400_{x+1} に演算器 120_{x+1} を選択させる。また、命令デコーダ 320_{x+1} は、ステージラッチ回路 310_{x+1} からの命令 $INST_{x+1}$ を解読デコードする。これにより、演算器 120_{x+1} は、ステージラッチ回路 110_{x+1} の値を用いて、命令 $INST_{x+1}$ に応じた演算を行う。つぎのステップSC22では、演算器 120_{x+1} の演算結果がステージラッチ回路 110_{x+2} に保持される。

【0106】

この場合、ステップSC23では、命令デコーダ 320_{x+1} は、マルチプレクサ 400_{x+1} にステージラッチ回路 110_{x+1} を選択させる。これにより、ステップSC24では、演算器 120_{x+1} が動作せず、ステップSC25では、ステージラッチ回路 110_{x+1} の値（演算器 120_x の演算結果）が、バイパスライン B_{x+1} およびマルチプレクサ 400_{x+1} を介してステージラッチ回路 110_{x+2} に保持される。以後、第 $x+2$ 演算ステージ～第 $n-1$ 演算ステージ（いずれも図示略）においては、第 $x+1$ 演算ステージの場合と同様にして、マルチプレ

クサによりステージラッチ回路が選択されることで、演算器 120_x の演算結果がそれぞれの演算ステージをスルーで順次通過する。また、第 $x+2$ 演算ステージ～第 $n-1$ 演算ステージにおいては、命令 $INST_x$ がステージラッチ回路に順次保持される。

【0107】

そして、第 n 演算ステージにおいて、ステップ SC 26（図 8 参照）では、命令デコーダ 320_n は、ステージラッチ回路 310_n に命令 $INST_x$ が保持されているため、ステップ SC 29 へ進む。なお、ステージラッチ回路 310_n に命令 $INST_n$ が保持されている場合、命令デコーダ 320_n は、ステップ SC 27 へ進み、マルチプレクサ 400_n に演算器 120_n を選択させる。また、命令デコーダ 320_n は、ステージラッチ回路 310_n からの命令 $INST_n$ をデコードする。これにより、演算器 120_n は、ステージラッチ回路 110_n の値を用いて、命令 $INST_n$ に応じた演算を行う。つぎのステップ SC 28 では、演算器 120_n の演算結果がステージラッチ回路 110_{n+1} に保持される。

【0108】

この場合、ステップ SC 29 では、命令デコーダ 320_n は、マルチプレクサ 400_n にステージラッチ回路 110_n を選択させる。これにより、ステップ SC 30 では、演算器 120_n が動作せず、ステップ SC 31 では、ステージラッチ回路 110_n の値（演算器 120_x の演算結果）が、バイパスライン B_n およびマルチプレクサ 400_n を介してステージラッチ回路 110_{n+1} に保持される。つぎのステップ SC 32 では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_x の演算結果が保持される。

【0109】

つぎに、第 $x+1$ 演算ステージの演算器 120_{x+1} に対して命令 $INST_{x+1}$ が発行された場合の動作について説明する。図 6 に示したステップ SC 1 では、被演算データ用レジスタ 100 、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第 1 被演算データ $SOURCE_1$ 、第 2 被演算データ $SOURCE_2$ および命令 $INST_{x+1}$ がそれぞれ保持される。これにより、上記第 1 被演算データ $SOURCE_1$ 、第 2 被演算データ $SOURCE_2$ および命令 $INST$

T_{x+1} がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 に保持される。

【0110】

以後、第1演算ステージ～第 $x-1$ 演算ステージにおいては、前述した動作（ステップSC2、ステップSC5～ステップSC8、ステップSC11～ステップSC13）と同様にして、スルー命令 SRC_1 およびスルー命令 SRC_2 がそれぞれデコードされる。つまり、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ がそれぞれの演算ステージをスルーで順次通過する。また、第1演算ステージ～第 $x-1$ 演算ステージにおいては、命令 $INST_{x+1}$ がステージラッチ回路に順次保持される。これにより、第 x 演算ステージにおいて、ステージラッチ回路 110_x 、ステージラッチ回路 210_x およびステージラッチ回路 310_x には、第1被演算データ $SOURCE_1$ 、第2被演算データ $SOURCE_2$ および命令 $INST_{x+1}$ がそれぞれ保持される。

【0111】

第 x 演算ステージにおいて、ステップSC14（図7参照）では、命令デコーダ 320_x は、ステージラッチ回路 310_x に命令 $INST_{x+1}$ が保持されているため、ステップSC17へ進む。ステップSC17では、命令デコーダ 320_x は、命令 $INST_{x+1}$ をスルー命令 SRC_1 に変換した後、ステップSC18へ進む。ステップSC18では、命令デコーダ 320_x は、演算器 120_x に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_x は、ステージラッチ回路 110_x の値をスルーで通過させる。つぎのステップSC19では、ステージラッチ回路 110_x の値がステージラッチ回路 110_{x+1} に保持される。このとき、ステージラッチ回路 310_x の値（命令 $INST_{x+1}$ ）がステージラッチ回路 310_{x+1} に保持される。

【0112】

つぎの第 $x+1$ 演算ステージにおいて、ステップSC20では、命令デコーダ 320_{x+1} は、ステージラッチ回路 310_{x+1} に命令 $INST_{x+1}$ が保持されているため、ステップSC21へ進み、マルチプレクサ 400_{x+1} に演算器 120_{x+1} を選択させる。また、命令デコーダ 320_{x+1} は、ステージラッチ回路 310_{x+1}

0_{x+1} からの命令 $INST_{x+1}$ をデコードする。これにより、演算器 120_{x+1} は、ステージラッチ回路 110_{x+1} の値を用いて、命令 $INST_{x+1}$ に応じた演算を行う。つぎのステップ SC 2 2 では、演算器 120_{x+1} の演算結果がステージラッチ回路 110_{x+2} に保持される。

【0 1 1 3】

以後、第 $x+2$ 演算ステージ～第 $n-1$ 演算ステージ（いずれも図示略）においては、ステップ SC 2 3～ステップ SC 2 5 と同様の動作が行われ、マルチプレクサによりステージラッチ回路が選択されることで、第 $x+1$ 演算ステージの演算器 120_{x+1} の演算結果がそれぞれの演算ステージをスルーで順次通過する。また、第 $x+2$ 演算ステージ～第 $n-1$ 演算ステージにおいては、命令 $INST_{x+1}$ がステージラッチ回路に順次保持される。

【0 1 1 4】

そして、第 n 演算ステージにおいて、ステップ SC 2 6（図 8 参照）では、命令デコーダ 320_n は、ステージラッチ回路 310_n に命令 $INST_{x+1}$ が保持されているため、ステップ SC 2 9 へ進む。ステップ SC 2 9 では、命令デコーダ 320_n は、マルチプレクサ 400_n にステージラッチ回路 110_n を選択させる。これにより、ステップ SC 3 0 では、演算器 120_n が動作せず、ステップ SC 3 1 では、ステージラッチ回路 110_n の値（演算器 120_{x+1} の演算結果）が、バイパスライン B_n およびマルチプレクサ 400_n を介してステージラッチ回路 110_{n+1} に保持される。つぎのステップ SC 3 2 では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_{x+1} の演算結果が保持される。

【0 1 1 5】

以上説明したように、実施の形態 3 によれば、第 x 演算ステージにおける演算実行時に、第 1 演算ステージ～第 $x-1$ 演算ステージにおいてスルー命令 SRC_1 およびスルー命令 SRC_2 をそれぞれデコードすることで第 1 被演算データ $SOURCE_1$ および第 2 被演算データ $SOURCE_2$ をスルーさせるようにしたので、第 1 演算ステージ～第 $x-1$ 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する

【0 1 1 6】

(実施の形態 4)

図 9 は、本発明にかかる実施の形態 4 の構成を示すブロック図である。この図においては、図 5 の各部に対応する部分には同一の符号をそれぞれ付ける。この図に示したパイプライン演算装置は、第 1 演算ステージ～第 n 演算ステージという n 段の演算ステージを有している。ただし、図 9 には、第 1 演算ステージ、第 $x-1$ ($1 < x < n$) 演算ステージ、第 x 演算ステージ、第 $x+1$ 演算ステージおよび第 n 演算ステージが図示されており、他の演算ステージの図示が省略されている。

【0 1 1 7】

また、図 9 に示した命令デコーダ 320_{11} 、命令デコーダ 320_{12} 等の機能は、後述するように、図 5 に示した命令デコーダ 320_{11} 、命令デコーダ 320_{12} 等の機能とは異なる。図 9 においては、第 1 演算ステージにマルチプレクサ 700_1 、マルチプレクサ 800_1 、バイパスライン B_{11} 、バイパスライン B_{12} が新たに設けられている。同様にして、図 9 においては、第 $x-1$ 演算ステージにマルチプレクサ 700_{x-1} 、マルチプレクサ 800_{x-1} 、バイパスライン B_{x-11} 、バイパスライン B_{x-12} が新たに設けられている。また、図 9 においては、第 $x+1$ 演算ステージに、図 5 に示したマルチプレクサ 400_{x+1} およびバイパスライン B_{x+1} が設けられておらず、さらに第 n 演算ステージに、図 5 に示したマルチプレクサ 400_n およびバイパスライン B_n が設けられていない。

【0 1 1 8】

図 9 において、第 1 演算ステージの命令デコーダ 320_{11} は、ステージラッチ回路 310_1 からの命令 $INST_{11}$ をデコードする。演算器 120_1 は、命令 $INST_{11}$ がデコードされたとき、ステージラッチ回路 110_1 の値（第 1 被演算データ $SOURCE_1$ ）を用いて、命令 $INST_{11}$ に対応する演算を行う。バイパスライン B_{11} は、演算器 120_1 に対して並列的に設けられており、ステージラッチ回路 110_1 の値を、演算器 120_1 を経由させることなく、マルチプレクサ 700_1 に導く役目をしている。

【0 1 1 9】

マルチプレクサ 700_1 は、命令デコーダ 320_{11} により切り替え制御される 2 入力 1 出力型の切替器であり、演算器 120_1 の演算結果、ステージラッチ回路 110_1 の値のうちいずれか一方を出力する。具体的には、マルチプレクサ 700_1 は、命令デコーダ 320_{11} により命令 $INST_{11}$ がデコードされたとき演算器 120_1 の演算結果を出力する。一方、マルチプレクサ 700_1 は、命令 $INST_{11}$ 以外の命令が命令デコーダ 320_{11} に入力されたとき、ステージラッチ回路 110_1 の値を出力する。

【0 1 2 0】

命令デコーダ 320_{12} は、ステージラッチ回路 310_1 からの命令 $INST_{12}$ をデコードする。演算器 220_1 は、命令 $INST_{12}$ がデコードされたとき、ステージラッチ回路 210_1 の値（第 2 被演算データ $SOURCE_2$ ）を用いて、命令 $INST_{12}$ に対応する演算を行う。バイパスライン B_{12} は、演算器 220_1 に対して並列的に設けられており、ステージラッチ回路 210_1 の値を、演算器 220_1 を経由させることなく、マルチプレクサ 800_1 に導く役目をしている。

【0 1 2 1】

マルチプレクサ 800_1 は、命令デコーダ 320_{12} により切り替え制御される 2 入力 1 出力型の切替器であり、演算器 220_1 の演算結果、ステージラッチ回路 210_1 の値のうちいずれか一方を出力する。具体的には、マルチプレクサ 800_1 は、命令デコーダ 320_{12} により命令 $INST_{12}$ がデコードされたとき演算器 220_1 の演算結果を出力する。一方、マルチプレクサ 800_1 は、命令 $INST_{12}$ 以外の命令が命令デコーダ 320_{12} に入力されたとき、ステージラッチ回路 210_1 の値を出力する。ステージラッチ回路 110_2 、ステージラッチ回路 210_2 およびステージラッチ回路 310_2 は、マルチプレクサ 700_1 、マルチプレクサ 800_1 およびステージラッチ回路 310_1 の出力値をそれぞれ保持する。以下、第 2 演算ステージ（図示略）～第 $x-1$ 演算ステージまでは、上述した第 1 演算ステージと同一構成とされている。

【0 1 2 2】

すなわち、第 $x-1$ 演算ステージにおいて、命令デコーダ 320_{x-11} は、ステージラッチ回路 310_{x-1} からの命令 $INST_{x-11}$ をデコードする。演算器 120_{x-1} は、命令 $INST_{x-11}$ がデコードされたとき、ステージラッチ回路 110_{x-1} の値を用いて、命令 $INST_{x-11}$ に対応する演算を行う。バイパスライン B_{x-11} は、演算器 120_{x-1} を経由させることなく、ステージラッチ回路 110_{x-1} の値をマルチプレクサ 700_{x-1} に導く役目をしている。

【0123】

マルチプレクサ 700_{x-1} は、命令デコーダ 320_{x-11} により切り替え制御され、命令デコーダ 320_{x-11} により命令 $INST_{x-11}$ がデコードされたとき演算器 120_{x-1} の演算結果を出力する。一方、マルチプレクサ 700_{x-1} は、命令 $INST_{x-11}$ 以外の命令が命令デコーダ 320_{x-11} に入力されたとき、ステージラッチ回路 110_{x-1} の値を出力する。

【0124】

命令デコーダ 320_{x-12} は、ステージラッチ回路 310_{x-1} からの命令 $INST_{x-12}$ をデコードする。演算器 220_{x-1} は、命令 $INST_{x-12}$ がデコードされたとき、ステージラッチ回路 210_{x-1} の値を用いて、命令 $INST_{x-12}$ に対応する演算を行う。バイパスライン B_{x-12} は、演算器 220_{x-1} を経由させることなく、ステージラッチ回路 210_{x-1} の値をマルチプレクサ 800_{x-1} に導く役目をしている。マルチプレクサ 800_{x-1} は、命令デコーダ 320_{x-12} により命令 $INST_{x-12}$ がデコードされたとき演算器 220_{x-1} の演算結果を出力し、命令 $INST_{x-12}$ 以外の命令が命令デコーダ 320_{x-12} に入力されたとき、ステージラッチ回路 210_{x-1} の値を出力する。

【0125】

つぎの第 x 演算ステージにおいて、ステージラッチ回路 110_x 、ステージラッチ回路 210_x 、およびステージラッチ回路 310_x は、マルチプレクサ 700_{x-1} の出力値、マルチプレクサ 800_{x-1} の出力値、およびステージラッチ回路 310_{x-1} の値をそれぞれ保持する。命令デコーダ 320_x は、ステージラッチ回路 310_x からの命令 $INST_x$ をデコードする。上記演算器 120_x は、命令 $INST_x$ がデコードされたとき、ステージラッチ回路 110_x の値、およ

びステージラッチ回路 210_x の値を用いて、命令 $INST_x$ に対応する演算を行い演算結果をステージラッチ回路 110_{x+1} へ出力する。また、命令デコーダ 320_x は、命令 $INST_x$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_x に対してデコードする。演算器 120_x は、スルー命令 SRC_1 がデコードされた場合、ステージラッチ回路 110_x の値、ステージラッチ回路 210_x の値のうち、ステージラッチ回路 110_x の値のみをスルーさせる。

【0126】

つぎの第 $x+1$ 演算ステージにおいて、ステージラッチ回路 110_{x+1} は、ステージラッチ回路 110_x の値または演算器 120_x の演算結果を保持する。ステージラッチ回路 310_{x+1} は、ステージラッチ回路 310_x の値を保持する。命令デコーダ 320_{x+1} は、ステージラッチ回路 310_{x+1} からの命令 $INST_{x+1}$ をデコードする。演算器 120_{x+1} は、命令 $INST_{x+1}$ がデコードされたとき、ステージラッチ回路 110_{x+1} の値を用いて、命令 $INST_{x+1}$ に対応する演算を行い演算結果をステージラッチ回路 110_{x+2} へ出力する。また、命令デコーダ 320_{x+1} は、命令 $INST_{x+1}$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_{x+1} に対してデコードする。演算器 120_{x+1} は、スルー命令 SRC_1 がデコードされた場合、ステージラッチ回路 110_{x+1} の値をスルーさせる。以下、第 $x+2$ 演算ステージ（図示略）～第 n 演算ステージまでは、上述した第 $x+1$ 演算ステージと同一構成とされている。

【0127】

すなわち、第 n 演算ステージにおいて、ステージラッチ回路 110_n は、第 $n-1$ 演算ステージ（図示略）のステージラッチ回路の値または演算器の演算結果を保持する。ステージラッチ回路 310_n は、第 $n-1$ 演算ステージのステージラッチ回路の値を保持する。命令デコーダ 320_n は、ステージラッチ回路 310_n からの命令 $INST_n$ をデコードする。演算器 120_n は、命令 $INST_n$ がデコードされたとき、ステージラッチ回路 110_n の値を用いて、命令 $INST_n$ に対応する演算を行い演算結果をステージラッチ回路 110_{n+1} へ出力する。

。また、命令デコーダ 320_n は、命令 $INST_n$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_n に対してデコードする。演算器 120_n は、スルー命令 SRC_1 がデコードされた場合、ステージラッチ回路 110_n の値をスルーさせる。演算結果用レジスタ 500 は、ステージラッチ回路 110_{n+1} の値、すなわち、パイプライン演算装置における演算結果（デスティネーション・オペランド）を保持する。

【0128】

また、実施の形態 4 において、第 1 演算ステージ～第 n 演算ステージまでのそれぞれの演算器は、実施の形態 3 と同様にして、互いに相関関係がない命令に基づいて演算をそれぞれ行う。つまり、相関関係がない場合には、それぞれの演算器における演算が単独で行われる。さらに、実施の形態 4 においては、第 $x+1$ 演算ステージ～第 n 演算ステージでスルー命令 SRC_1 がそれぞれデコードされることで、第 x 演算ステージにおける演算器 120_x の演算結果がスルーされる。

【0129】

つぎに、上述した実施の形態 4 の動作について図 10～図 12 に示したフローチャートを参照しつつ説明する。はじめに、命令 $INST_x$ が発行された場合の動作について説明する。図 10 に示したステップ SD 1 では、被演算データ用レジスタ 100 、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第 1 被演算データ $SOURCE_1$ 、第 2 被演算データ $SOURCE_2$ および命令 $INST_x$ がそれぞれ保持される。これにより、上記第 1 被演算データ $SOURCE_1$ 、第 2 被演算データ $SOURCE_2$ および命令 $INST_x$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 にそれぞれ保持される。

【0130】

つぎのステップ SD 2 では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステージラッチ回路 310_1 に命令 $INST_x$ が保持されているため、ステップ SD 5 へ進む。なお、ステージラッチ回路 310_1 に命令 $INST_{11}$ および命令 $INST_{12}$ が保持されている場合、命令デコーダ 320_{11} および命令デ

コード 320_{12} は、ステップ $SD3$ へ進み、マルチプレクサ 700_1 およびマルチプレクサ 800_1 に演算器 120_1 および演算器 220_1 を選択させる。

【0131】

また、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、命令 $INST_{11}$ および命令 $INST_{12}$ を演算器 120_1 および演算器 220_1 に対してそれぞれデコードする。これにより、演算器 120_1 および演算器 220_1 は、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_1 の値（第2被演算データ $SOURCE_2$ ）を用いて、命令 $INST_{11}$ および命令 $INST_{12}$ に応じた演算を行う。つぎのステップ $SD4$ では、演算器 120_1 および演算器 220_1 の演算結果がステージラッチ回路 110_2 およびステージラッチ回路 210_2 にそれぞれ保持される。

【0132】

この場合、ステップ $SD5$ では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、マルチプレクサ 700_1 およびマルチプレクサ 800_1 にステージラッチ回路 110_1 およびステージラッチ回路 210_1 を選択させる。ステップ $SD6$ では、演算器 120_1 および演算器 220_1 が動作せず、ステップ $SD7$ では、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）が、バイパスライン B_{11} およびマルチプレクサ 700_1 を介してステージラッチ回路 110_2 に保持される。同様に、ステージラッチ回路 210_1 の値（第2被演算データ $SOURCE_2$ ）が、バイパスライン B_{12} およびマルチプレクサ 800_1 を介してステージラッチ回路 210_2 に保持される。

【0133】

以後、第2演算ステージ～第 $x-2$ 演算ステージ（いずれも図示略）においては、第1演算ステージの場合と同様に、マルチプレクサによりステージラッチ回路が選択されることで、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ がそれぞれの演算ステージをスルーで順次通過する。また、第2演算ステージ～第 $x-2$ 演算ステージにおいては、命令 $INST_x$ がステージラッチ回路に順次保持される。

【0134】

そして、第 $x-1$ 演算ステージにおいて、ステップSD 8では、命令デコーダ 320_{x-11} および命令デコーダ 320_{x-12} は、ステージラッチ回路 310_{x-1} に命令 $INST_x$ が保持されているため、ステップSD 11へ進む。なお、ステージラッチ回路 310_{x-1} に命令 $INST_{x-11}$ および命令 $INST_{x-12}$ が保持されている場合、命令デコーダ 320_{x-11} および命令デコーダ 320_{x-12} は、ステップSD 9へ進み、マルチプレクサ 700_{x-1} およびマルチプレクサ 800_{x-1} に演算器 120_{x-1} および演算器 220_{x-1} を選択させる。

【0135】

また、命令デコーダ 320_{x-11} および命令デコーダ 320_{x-12} は、命令 $INST_{x-11}$ および命令 $INST_{x-12}$ を演算器 120_{x-1} および演算器 220_{x-1} に対してそれぞれデコードする。これにより、演算器 120_{x-1} および演算器 220_{x-1} は、ステージラッチ回路 110_{x-1} の値およびステージラッチ回路 210_{x-1} の値を用いて、命令 $INST_{x-11}$ および命令 $INST_{x-12}$ に応じた演算を行う。つぎのステップSD 10では、演算器 120_{x-1} および演算器 220_{x-1} の演算結果がステージラッチ回路 110_x およびステージラッチ回路 210_x にそれぞれ保持される。

【0136】

この場合、ステップSD 11では、命令デコーダ 320_{x-11} および命令デコーダ 320_{x-12} は、マルチプレクサ 700_{x-1} およびマルチプレクサ 800_{x-1} にステージラッチ回路 110_{x-1} およびステージラッチ回路 210_{x-1} を選択させる。ステップSD 12では、演算器 120_{x-1} および演算器 220_{x-1} が動作せず、ステップSD 13では、ステージラッチ回路 110_{x-1} の値（第1被演算データ $SOURCE_1$ ）が、バイパスライン B_{x-11} およびマルチプレクサ 700_{x-1} を介してステージラッチ回路 110_x に保持される。同様にして、ステージラッチ回路 210_{x-1} の値（第2被演算データ $SOURCE_2$ ）が、バイパスライン B_{x-12} およびマルチプレクサ 800_{x-1} を介してステージラッチ回路 110_x に保持される。このとき、ステージラッチ回路 310_{x-1} の値がステージラッチ回路 310_x に保持される。

【0137】

つぎの第 x 演算ステージにおいて、ステップSD14（図11参照）では、命令デコーダ 320_x は、ステージラッチ回路 310_x に命令 $INST_x$ が保持されているため、ステップSD15へ進む。なお、ステージラッチ回路 310_x に命令 $INST_x$ 以外の命令が保持されている場合、命令デコーダ 320_x は、ステップSD17へ進み、命令 $INST_x$ 以外の命令をスルー命令 SRC_1 に変換した後、ステップSD18へ進む。ステップSD18では、命令デコーダ 320_x は、演算器 120_x に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_x は、ステージラッチ回路 110_x の値をスルーで通過させる。つぎのステップSD19では、ステージラッチ回路 110_x の値がステージラッチ回路 110_{x+1} に保持される。

【0138】

この場合、ステップSD15では、命令デコーダ 320_x は、命令 $INST_x$ を演算器 120_x に対してデコードする。これにより、演算器 120_x はステージラッチ回路 110_x の値（第1被演算データ $SOURCE_1$ ）、およびステージラッチ回路 210_x の値（第2被演算データ $SOURCE_2$ ）を用いて、命令 $INST_x$ に応じた演算を行う。これにより、ステップSD16では、演算器 120_x の演算結果がステージラッチ回路 110_{x+1} に保持される。このとき、ステージラッチ回路 310_x の値（命令 $INST_x$ ）がステージラッチ回路 310_{x+1} に保持される。

【0139】

つぎの第 $x+1$ 演算ステージにおいて、ステップSD20では、命令デコーダ 320_{x+1} は、ステージラッチ回路 310_{x+1} に命令 $INST_x$ が保持されているため、ステップSD23へ進む。なお、ステージラッチ回路 310_{x+1} に命令 $INST_{x+1}$ が保持されている場合、命令デコーダ 320_{x+1} は、ステップSD21へ進み、演算器 120_{x+1} に対して命令 $INST_{x+1}$ をデコードする。これにより、演算器 120_{x+1} は、ステージラッチ回路 110_{x+1} の値を用いて、命令 $INST_{x+1}$ に応じた演算を行う。つぎのステップSD22では、演算器 120_{x+1} の演算結果がステージラッチ回路 110_{x+2} に保持される。

【0140】

この場合、ステップSD23では、命令デコーダ320_{x+1}は、INST_{x+1}以外の命令（命令INST_x）をスルー命令SRC₁に変換した後、ステップSD24へ進む。ステップSD24では、演算器120_{x+1}に対してスルー命令SRC₁をデコードする。これにより、演算器120_{x+1}は、ステージラッチ回路110_{x+1}の値（演算器120_xの演算結果）をスルーで通過させる。つぎのステップSD25では、ステージラッチ回路110_{x+1}の値がステージラッチ回路110_{x+2}に保持される。このとき、ステージラッチ回路310_{x+1}の値がステージラッチ回路310_{x+2}に保持される。以後、第x+2演算ステージ～第n-1演算ステージ（いずれも図示略）においては、第x+1演算ステージの場合と同様にして、スルー命令SRC₁がデコードされることで、演算器120_xの演算結果がそれぞれの演算ステージをスルーで順次通過する。また、第x+2演算ステージ～第n-1演算ステージにおいては、命令INST_xがステージラッチ回路に順次保持される。

【0141】

そして、第n演算ステージにおいて、ステップSD26（図12参照）では、命令デコーダ320_nは、ステージラッチ回路310_nに命令INST_xが保持されているため、ステップSD29へ進む。なお、ステージラッチ回路310_nに命令INST_nが保持されている場合、命令デコーダ320_nは、ステップSD27へ進み、演算器120_nに対して命令INST_nをデコードする。これにより、演算器120_nは、ステージラッチ回路110_nの値を用いて、命令INST_nに応じた演算を行う。つぎのステップSD28では、演算器120_nの演算結果がステージラッチ回路110_{n+1}に保持される。

【0142】

この場合、ステップSD29では、命令デコーダ320_nは、INST_n以外の命令をスルー命令SRC₁に変換した後、ステップSD30へ進む。ステップSD30では、命令デコーダ320_nは、演算器120_nに対してスルー命令SRC₁をデコードする。これにより、演算器120_nは、ステージラッチ回路110_nの値（演算器120_xの演算結果）をスルーで通過させる。つぎのステップSD31では、ステージラッチ回路110_nの値がステージラッチ回路110

$n+1$ に保持される。つぎのステップSD32では、演算結果用レジスタ500には、パイプライン演算装置の演算結果として演算器 120_x の演算結果が保持される。

【0143】

つぎに第 $x+1$ 演算ステージの演算器 120_{x+1} に対して命令 $INST_{x+1}$ が発行された場合の動作について説明する。図10に示したステップSD1では、被演算データ用レジスタ100、被演算データ用レジスタ200および命令用レジスタ300に、第1被演算データ $SOURCE_1$ 、第2被演算データ $SOURCE_2$ および命令 $INST_{x+1}$ がそれぞれ保持される。これにより、上記第1被演算データ $SOURCE_1$ 、第2被演算データ $SOURCE_2$ および命令 $INST_{x+1}$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 にそれぞれ保持される。

【0144】

以後、第1演算ステージ～第 $x-1$ 演算ステージにおいては、前述した動作（ステップSD2、ステップSD5～ステップSD8、ステップSD11～ステップSD13）と同様にして、マルチプレクサにおいてステージラッチ回路が選択されることで、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ がそれぞれの演算ステージをスルーで順次通過する。また、第1演算ステージ～第 $x-1$ 演算ステージにおいては、命令 $INST_{x+1}$ がステージラッチ回路に順次保持される。これにより、第 x 演算ステージにおいて、ステージラッチ回路 110_x 、ステージラッチ回路 210_x およびステージラッチ回路 310_x には、第1被演算データ $SOURCE_1$ 、第2被演算データ $SOURCE_2$ および命令 $INST_{x+1}$ がそれぞれ保持されている。

【0145】

第 x 演算ステージにおいて、ステップSD14（図11参照）では、命令デコーダ 320_x は、ステージラッチ回路 310_x に命令 $INST_{x+1}$ が保持されているため、ステップSD17へ進む。ステップSD17では、命令デコーダ 320_x は、命令 $INST_{x+1}$ をスルー命令 SRC_1 に変換した後、ステップSD18へ進む。ステップSD18では、命令デコーダ 320_x は、演算器 120_x に

対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_x は、ステージラッチ回路 110_x の値をスルーで通過させる。つぎのステップ $SD19$ では、ステージラッチ回路 110_x の値がステージラッチ回路 110_{x+1} に保持される。

【0146】

つぎの第 $x+1$ 演算ステージにおいて、ステップ $SD20$ では、命令デコーダ 320_{x+1} は、ステージラッチ回路 310_{x+1} に命令 $INST_{x+1}$ が保持されているため、ステップ $SD21$ へ進み、演算器 120_{x+1} に対して命令 $INST_{x+1}$ をデコードする。これにより、演算器 120_{x+1} は、ステージラッチ回路 110_{x+1} の値を用いて、命令 $INST_{x+1}$ に応じた演算を行う。つぎのステップ $SD22$ では、演算器 120_{x+1} の演算結果がステージラッチ回路 110_{x+2} に保持される。

【0147】

そして、第 n 演算ステージにおいて、ステップ $SD26$ (図12参照) では、命令デコーダ 320_n は、ステージラッチ回路 310_n に命令 $INST_{x+1}$ が保持されているため、ステップ $SD29$ へ進む。ステップ $SD29$ では、命令デコーダ 320_n は、命令 $INST_{x+1}$ をスルー命令 SRC_1 に変換した後、ステップ $SD30$ へ進む。ステップ $SD30$ では、命令デコーダ 320_n は、演算器 120_n に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_n は、ステージラッチ回路 110_n の値 (演算器 120_{x+1} の演算結果) をスルーで通過させる。つぎのステップ $SD31$ では、ステージラッチ回路 110_n の値がステージラッチ回路 110_{n+1} に保持される。つぎのステップ $SD32$ では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_{x+1} の演算結果が保持される。

【0148】

以上説明したように、実施の形態4によれば、第 x 演算ステージにおける演算実行時に、第 $x+1$ 演算ステージ～第 n 演算ステージにおいてスルー命令 SRC_1 をそれぞれデコードすることで第 x 演算ステージにおける演算器 120_x の演算結果をスルーさせるようにしたので、第 $x+1$ 演算ステージ～第 n 演算ステー

ジにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する。

【0 1 4 9】

(実施の形態 5)

さて、前述した実施の形態 1 においては、図 1 に示した第 1 演算ステージの演算器 120_1 と、第 2 演算ステージの演算器 120_2 とが、互いに相関関係がない命令（命令 $INST_1$ 、命令 $INST_2$ ）に基づいて演算をそれぞれ行う場合について説明したが、相関関係がある命令に基づいて演算を行う複数の演算器を備えるようにしてもよい。ここで、相関関係がある命令とは、演算器 120_1 および演算器 120_2 の双方に対して発行される命令であって、演算器 120_1 の演算結果を用いて、演算器 120_2 において演算を行わせるための命令をいう。以下においては、この場合を実施の形態 5 として詳述する。

【0 1 5 0】

図 1 3 は、本発明にかかる実施の形態 5 の構成を示すブロック図である。この図において、図 1 の各部にそれぞれ対応する部分には同一の符号を付け、その説明を省略する。ただし、図 1 3 に示した第 1 演算ステージの命令デコーダ 320_1 および演算器 120_1 、第 2 演算ステージの命令デコーダ 320_2 および演算器 120_2 のそれぞれの機能は、これらに対応する図 1 に示したものの機能と異なる。

【0 1 5 1】

すなわち、図 1 3 に示した第 1 演算ステージにおいて、命令デコーダ 320_1 は、ステージラッチ回路 310_1 からの命令 $INST_{12}$ をデコードする。ここで命令 $INST_{12}$ は、演算器 120_1 および演算器 120_2 の双方に対して発行される命令であって、演算器 120_1 の演算結果を用いて、演算器 120_2 において演算を行わせるための命令である。つまり、命令 $INST_{12}$ は、演算器 120_1 と演算器 120_2 との間で、演算に関して相関関係を持たせるための命令である。また、演算器 120_1 は、命令 $INST_{12}$ がデコードされたとき、ステージラッチ回路 110_1 の値（第 1 被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_1 の値（第 2 被演算データ）を用いて、命令 $INST_{12}$ に対応す

る演算を行い演算結果をステージラッチ回路 110_2 へ出力する。

【0152】

また、命令デコーダ 320_1 は、命令 $INST_{12}$ 以外の命令（この場合、命令 $INST_2$ ）が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_1 に対してデコードする。演算器 120_1 は、上記スルー命令 SRC_1 がデコードされた場合、二つの被演算データ（第1被演算データ $SOURCE_1$ 、第2被演算データ）のうち、第1被演算データ $SOURCE_1$ のみをスルーさせる。

【0153】

第2演算ステージにおいて、命令デコーダ 320_2 は、上述した命令 $INST_{12}$ に加えて、命令 $INST_2$ のデコードも行う。すなわち、命令デコーダ 320_2 は、ステージラッチ回路 310_2 からの上述した命令 $INST_{12}$ をデコードする。演算器 120_2 は、命令 $INST_{12}$ がデコードされたとき、ステージラッチ回路 110_2 の値（演算器 120_1 の演算結果）を用いて、命令 $INST_{12}$ に対応する演算を行い演算結果を出力する。

【0154】

また、命令デコーダ 320_2 は、ステージラッチ回路 310_2 からの命令 $INST_2$ をデコードする。ここで、命令 $INST_2$ は、演算器 120_1 とは独立して演算器 120_2 において演算を行わせるための命令である。演算器 120_2 は、命令 $INST_2$ がデコードされたとき、ステージラッチ回路 110_2 の値（第1被演算データ $SOURCE_1$ ）を用いて、命令 $INST_2$ に対応する演算を行い演算結果を出力する。

【0155】

マルチプレクサ 400_2 は、命令デコーダ 320_2 により命令 $INST_{12}$ または命令 $INST_2$ がデコードされたとき演算器 120_2 の演算結果を出力する。一方、マルチプレクサ 400_2 は、命令 $INST_{12}$ および命令 $INST_2$ 以外の命令が命令デコーダ 320_2 に入力されたとき、ステージラッチ回路 110_2 の値を出力する。ステージラッチ回路 110_3 は、マルチプレクサ 400_2 の切り替え状態に応じて、演算器 120_2 の演算結果またはステージラッチ回路 110

2 の値を保持する。演算結果用レジスタ 500 は、ステージラッチ回路 110₃ の値、すなわち、パイプライン演算装置における演算結果（デスティネーション・オペランド）を保持する。

【0156】

つぎに、上述した実施の形態 5 の動作について図 14 に示したフローチャートを参照しつつ説明する。はじめに、命令 $INST_{12}$ が発行された場合の動作について説明する。図 14 に示したステップ SE1 では、被演算データ用レジスタ 100、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第 1 被演算データ $SOURCE_1$ 、第 2 被演算データおよび命令 $INST_{12}$ がそれぞれ保持される。これにより、上記第 1 被演算データ $SOURCE_1$ 、第 2 被演算データおよび命令 $INST_{12}$ がステージラッチ回路 110₁、ステージラッチ回路 210₁ およびステージラッチ回路 310₁ にそれぞれ保持される。

【0157】

つぎのステップ SE2 では、命令デコーダ 320₁ は、ステージラッチ回路 310₁ に保持されている命令の種類を判断し、この場合、命令 $INST_{12}$ が保持されているため、ステップ SE3 へ進む。ステップ SE3 では、命令デコーダ 320₁ は、ステージラッチ回路 310₁ からの命令 $INST_{12}$ をデコードする。これにより、演算器 120₁ はステージラッチ回路 110₁ の値（第 1 被演算データ $SOURCE_1$ ）およびステージラッチ回路 210₁ の値（第 2 被演算データ）を用いて、命令 $INST_{12}$ に応じた演算を行う。

【0158】

つぎのステップ SE4 では、演算器 120₁ の演算結果がステージラッチ回路 110₂ に保持される。このとき、ステージラッチ回路 310₁ の値（命令 $INST_{12}$ ）がステージラッチ回路 310₂ に保持される。つぎのステップ SE8 では、命令デコーダ 320₂ は、ステージラッチ回路 310₂ に命令 $INST_{12}$ が保持されているため、ステップ SE9 へ進む。なお、ステージラッチ回路 310₂ に命令 $INST_{12}$ および命令 $INST_2$ 以外の命令が保持されている場合、命令デコーダ 320₂ は、ステップ SE11 へ進む。ステップ SE11 では、命令デコーダ 320₂ は、マルチプレクサ 400₂ にステージラッチ回路 110₂ を

選択させる。これにより、ステップSE12では、演算器 120_2 が動作せず、ステップSE13では、ステージラッチ回路 110_2 の値が、パイプラインB $_2$ およびマルチプレクサ 400_2 を介してステージラッチ回路 110_3 に保持される。

【0159】

この場合、ステップSE9では、命令デコーダ 320_2 は、マルチプレクサ 400_2 に演算器 120_2 を選択させる。また、命令デコーダ 320_2 は、ステージラッチ回路 310_2 からの命令 $INST_{12}$ をデコードする。これにより、演算器 120_2 は、ステージラッチ回路 110_2 の値（演算器 120_1 の演算結果）を用いて、命令 $INST_{12}$ に応じた演算を行う。つぎのステップSE10では、演算器 120_2 の演算結果がステージラッチ回路 110_3 に保持される。つぎのステップSE14では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_2 の演算結果が保持される。

【0160】

つぎに、命令 $INST_2$ が発行された場合の動作について説明する。図14に示したステップSE1では、被演算データ用レジスタ 100 、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第1被演算データ $SOURCE_1$ 、第2被演算データおよび命令 $INST_2$ がそれぞれ保持される。これにより、上記第1被演算データ $SOURCE_1$ 、第2被演算データおよび命令 $INST_2$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 にそれぞれ保持される。

【0161】

つぎのステップSE2では、命令デコーダ 320_1 は、ステージラッチ回路 310_1 に命令 $INST_2$ が保持されているため、ステップSE5へ進む。ステップSE5では、命令デコーダ 320_1 は、ステージラッチ回路 310_1 の値（命令 $INST_2$ ）を解読し、命令 $INST_2$ をスルー命令 SRC_1 に変換した後、ステップSE6へ進む。ステップSE6では、命令デコーダ 320_1 は、演算器 120_1 に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_1 は、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）

をスルーで通過させる。

【0162】

つぎのステップSE7では、ステージラッチ回路 110_1 の値（第1被演算データ $SOURCE_1$ ）がステージラッチ回路 110_2 に保持される。このとき、ステージラッチ回路 310_1 の値（命令 $INST_2$ ）がステージラッチ回路 310_2 に保持される。つぎのステップSE8では、命令デコーダ 320_2 は、ステージラッチ回路 310_2 に命令 $INST_2$ が保持されているため、ステップSE9へ進む。

【0163】

ステップSE9では、命令デコーダ 320_2 は、マルチプレクサ 400_2 に演算器 120_2 を選択させる。また、命令デコーダ 320_2 は、ステージラッチ回路 310_2 からの命令 $INST_2$ をデコードする。これにより、演算器 120_2 は、ステージラッチ回路 110_2 の値（第1被演算データ $SOURCE_1$ ）を用いて、命令 $INST_2$ に応じた演算を行う。つぎのステップSE10では、演算器 120_2 の演算結果がステージラッチ回路 110_3 に保持される。つぎのステップSE14では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_2 の演算結果が保持される。

【0164】

以上説明したように、実施の形態5によれば、命令デコーダ 320_1 において命令 $INST_{12}$ および命令 $INST_2$ 以外の命令をスルー命令 SRC_1 に変換し、演算器 120_1 で第1被演算データ $SOURCE_1$ をスルーさせるとともに、第2演算ステージにバイパスライン B_2 を設けたので、演算器 120_1 と演算器 120_2 との間でステージラッチ回路 110_2 の共有、および配線量の低減が可能となる。したがって、実施の形態5によれば、ステージラッチ回路 110_2 の共有化により、ハードウェア量および消費電力が低減する。また、実施の形態5によれば、第2演算ステージの演算器 120_2 において、第1演算ステージの演算器 120_1 とは独立して演算を行うことができる。

【0165】

（実施の形態6）

さて、前述した実施の形態 2 においては、図 3 に示した第 1 演算ステージの演算器 120_1 と、第 2 演算ステージの演算器 120_2 とが、互いに相関関係がない命令（命令 $INST_1$ 、命令 $INST_2$ ）に基づいて演算をそれぞれ行う場合について説明したが、実施の形態 5 と同様にして、相関関係がある命令に基づいて演算を行う複数の演算器を備えるようにしてもよい。以下においては、この場合を実施の形態 6 として詳述する。

【0166】

図 15 は、本発明にかかる実施の形態 6 の構成を示すブロック図である。この図において、図 3 の各部にそれぞれ対応する部分には同一の符号を付け、その説明を省略する。ただし、図 15 に示した第 1 演算ステージの命令デコーダ 320_1 および演算器 120_1 、第 2 演算ステージの命令デコーダ 320_2 および演算器 120_2 のそれぞれの機能は、これらに対応する図 3 に示したものの機能と異なる。

【0167】

すなわち、図 15 に示した第 1 演算ステージにおいて、命令デコーダ 320_1 は、ステージラッチ回路 310_1 からの命令 $INST_{12}$ をデコードする。ここで命令 $INST_{12}$ は、演算器 120_1 および演算器 120_2 の双方に対して発行される命令であって、演算器 120_1 の演算結果を用いて、演算器 120_2 において演算を行わせるための命令である。つまり、命令 $INST_{12}$ は、演算器 120_1 と演算器 120_2 との間で、演算に関して相関関係を持たせるための命令である。演算器 120_1 は、命令 $INST_{12}$ がデコードされたとき、ステージラッチ回路 110_1 の値（第 1 被演算データ）、およびステージラッチ回路 210_1 の値（第 2 被演算データ）を用いて、命令 $INST_{12}$ に対応する演算を行い演算結果 $RESULT_1$ をステージラッチ回路 110_2 へ出力する。

【0168】

また、命令デコーダ 320_1 は、上述した命令 $INST_{12}$ に加えて、命令 $INST_1$ のデコードも行う。ここで、命令 $INST_1$ は、演算器 120_2 とは独立して演算器 120_1 において演算を行わせるための命令である。すなわち、命令デコーダ 320_1 は、ステージラッチ回路 310_1 からの上述した命令 $INST$

1 をデコードする。演算器 120_1 は、命令 $INST_1$ がデコードされたとき、ステージラッチ回路 110_1 の値（第1被演算データ）、およびステージラッチ回路 210_1 の値（第2被演算データ）を用いて、命令 $INST_1$ に対応する演算を行い演算結果 $RESULT_1$ を出力する。

【0169】

マルチプレクサ 400_1 は、命令デコーダ 320_1 により切り替え制御され、命令デコーダ 320_1 により命令 $INST_{12}$ または命令 $INST_1$ がデコードされたとき演算器 120_1 の演算結果 $RESULT_1$ を出力する。一方、マルチプレクサ 400_1 は、命令 $INST_{12}$ および命令 $INST_1$ 以外の命令が命令デコーダ 320_1 に入力されたとき、ステージラッチ回路 110_1 の値を出力する。

【0170】

第2演算ステージにおいて、命令デコーダ 320_2 は、ステージラッチ回路 310_2 からの命令 $INST_{12}$ をデコードする。演算器 120_2 は、命令 $INST_{12}$ がデコードされたとき、ステージラッチ回路 110_2 の値（演算結果 $RESULT_1$ ）を用いて、命令 $INST_{12}$ に対応する演算を行い演算結果をステージラッチ回路 110_3 へ出力する。

【0171】

また、命令デコーダ 320_2 は、命令 $INST_{12}$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_2 に対してデコードする。演算器 120_2 は、上記スルー命令 SRC_1 がデコードされた場合、ステージラッチ回路 110_2 の値をスルーさせる。

【0172】

つぎに、上述した実施の形態5の動作について図16に示したフローチャートを参照しつつ説明する。はじめに、命令 $INST_{12}$ が発行された場合の動作について説明する。図16に示したステップSF1では、被演算データ用レジスタ 100 、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第1被演算データ、第2被演算データおよび命令 $INST_{12}$ がそれぞれ保持される。これにより、上記第1被演算データ、第2被演算データおよび命令 $INST_{12}$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回

路 3 1 0₁ にそれぞれ保持される。

【0 1 7 3】

つぎのステップ S F 2 では、命令デコーダ 3 2 0₁ は、ステージラッチ回路 3 1 0₁ に保持された命令の種類を判断し、この場合、ステージラッチ回路 3 1 0₁ に命令 I N S T₁₂ が保持されているため、ステップ S F 3 へ進む。なお、ステージラッチ回路 3 1 0₁ に命令 I N S T₁₂ および命令 I N S T₁ 以外の命令が保持されている場合、命令デコーダ 3 2 0₁ は、ステップ S F 5 へ進む。ステップ S F 5 では、命令デコーダ 3 2 0₁ は、マルチプレクサ 4 0 0₁ にステージラッチ回路 1 1 0₁ を選択させる。ステップ S F 6 では、演算器 1 2 0₁ が動作せず、ステップ S F 7 では、ステージラッチ回路 1 1 0₁ の値（第 1 被演算データ）が、バイパスライン B₁ およびマルチプレクサ 4 0 0₁ を介してステージラッチ回路 1 1 0₂ に保持される。

【0 1 7 4】

この場合、ステップ S F 3 では、命令デコーダ 3 2 0₁ は、マルチプレクサ 4 0 0₁ に演算器 1 2 0₁ を選択させる。また、命令デコーダ 3 2 0₁ は、命令 I N S T₁₂ を演算器 1 2 0₁ に対してデコードする。これにより、演算器 1 2 0₁ は、ステージラッチ回路 1 1 0₁ の値（第 1 被演算データ）、およびステージラッチ回路 2 1 0₁ の値（第 2 被演算データ）を用いて、命令 I N S T₁₂ に応じた演算を行う。つぎのステップ S F 4 では、演算器 1 2 0₁ の演算結果 R E S U L T₁ がマルチプレクサ 4 0 0₁ を介して、ステージラッチ回路 1 1 0₂ に保持される。これに並行して、ステージラッチ回路 3 1 0₁ の値（命令 I N S T₁₂）がステージラッチ回路 3 1 0₂ に保持される。

【0 1 7 5】

つぎのステップ S F 8 では、命令デコーダ 3 2 0₂ は、ステージラッチ回路 3 1 0₂ に保持された命令の種類を判断し、この場合、ステージラッチ回路 3 1 0₂ に命令 I N S T₁₂ が保持されているため、ステップ S F 9 へ進む。ステップ S F 9 では、命令デコーダ 3 2 0₂ は、演算器 1 2 0₂ に対して命令 I N S T₁₂ をデコードする。演算器 1 2 0₂ は、ステージラッチ回路 1 1 0₂ の値（演算結果 R E S U L T₁）を用いて、命令 I N S T₁₂ に応じた演算を実行する。つぎのス

ステップSF10では、演算器120₂の演算結果がステージラッチ回路110₃に保持され、ステップSF14では、演算結果用レジスタ500には、パイプライン演算装置の演算結果として演算器120₂の演算結果が保持される。

【0176】

つぎに、命令INST₁が発行された場合の動作について説明する。図16に示したステップSF1では、被演算データ用レジスタ100、被演算データ用レジスタ200および命令用レジスタ300に、第1被演算データ、第2被演算データおよび命令INST₁がそれぞれ保持される。これにより、上記第1被演算データ、第2被演算データおよび命令INST₁がステージラッチ回路110₁、ステージラッチ回路210₁およびステージラッチ回路310₁にそれぞれ保持される。

【0177】

つぎのステップSF2では、命令デコーダ320₁は、ステージラッチ回路310₁に命令INST₁が保持されているため、ステップSF3へ進み、マルチプレクサ400₁に演算器120₁を選択させる。また、命令デコーダ320₁は、命令INST₁を演算器120₁に対してデコードする。これにより、演算器120₁は、ステージラッチ回路110₁の値（第1被演算データ）、およびステージラッチ回路210₁の値（第2被演算データ）を用いて、命令INST₁に応じた演算を行う。つぎのステップSF4では、演算器120₁の演算結果RESULT₁がマルチプレクサ400₁を介して、ステージラッチ回路110₂に保持される。これに並行して、ステージラッチ回路310₁の値（命令INST₁）がステージラッチ回路310₂に保持される。

【0178】

つぎのステップSF8では、命令デコーダ320₂は、ステージラッチ回路310₂に命令INST₁が保持されているため、ステップSF11へ進む。ステップSF11では、命令デコーダ320₂は、ステージラッチ回路310₂の値（命令INST₁）を解釈し、命令INST₁をスルー命令SRC₁に変換した後、ステップSF12へ進む。ステップSF12では、命令デコーダ320₂は、演算器120₂に対してスルー命令SRC₁をデコードする。これにより、演

演算器 120_2 は、ステージラッチ回路 110_2 の値（演算結果 $RESULT_1$ ）をスルーで通過させる。つぎのステップ $SF13$ では、ステージラッチ回路 110_2 の値（演算結果 $RESULT_1$ ）がステージラッチ回路 110_3 に保持される。つぎのステップ $SF14$ では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_1 の演算結果 $RESULT_1$ が保持される。

【0179】

以上説明したように、実施の形態 6 によれば、第 1 演算ステージにマルチプレクサ 400_1 を設けて、ステージラッチ回路 110_1 の値または演算器 120_1 の演算結果のいずれかをステージラッチ回路 110_2 に保持させるとともに、命令デコーダ 320_2 において命令 $INST_{12}$ 以外の命令をスルー命令 SRC_1 に変換し、演算器 120_2 で演算器 120_1 の演算結果 $RESULT_1$ をスルーさせるようにしたので、演算器 120_2 の下流のステージラッチ回路および配線量の低減が可能となる。したがって、実施の形態 6 によれば、ハードウェア量および消費電力が低減する。また、実施の形態 6 によれば、第 1 演算ステージの演算器 120_1 において、第 2 演算ステージの演算器 120_2 とは独立して演算を行うことができる。

【0180】

（実施の形態 7）

さて、前述した実施の形態 3 において、図 5 に示した第 1 演算ステージ～第 n 演算ステージでは、互いに相関関係がない命令に基づいて演算をそれぞれ行う場合について説明したが、相関関係がある命令に基づいて演算を行う複数の演算器を備えるようにしてもよい。以下においては、この場合を実施の形態 7 として詳述する。

【0181】

図 17 および図 18 は、本発明にかかる実施の形態 7 の構成を示すブロック図である。この図において、図 5 の各部にそれぞれ対応する部分には同一の符号を付け、その説明を省略する。図 17 においては、図 5 に示した第 $x-1$ 演算ステージ～第 $x+1$ 演算ステージに代えて、第 $r-1$ 演算ステージ、第 r 演算ステー

ジ、第 $r + 1$ 演算ステージおよび第 s 演算ステージ（ただし、 $1 < r < s < n$ ）がそれぞれ図示されている。図 1 7 に示した第 $r - 1$ 演算ステージ、第 r 演算ステージおよび第 $r + 1$ 演算ステージ（第 s 演算ステージ）のそれぞれの構成は、図 5 に示した第 $x - 1$ 演算ステージ、第 x 演算ステージおよび第 $x + 1$ 演算ステージのそれぞれの構成と同様である。ただし、図 1 7 に示した第 $r - 1$ 演算ステージ、第 r 演算ステージおよび第 $r + 1$ 演算ステージにおける命令デコーダの機能が、図 5 に示した命令デコーダの機能と異なる。

【0 1 8 2】

すなわち、図 1 7 に示した第 $r - 1$ 演算ステージにおいて、ステージラッチ回路 $1 1 0_{r-1}$ は、第 1 被演算データ $SOURCE_1$ または第 $r - 2$ 演算ステージ（図示略）の演算器の演算結果を保持する。ステージラッチ回路 $2 1 0_{r-1}$ は、第 2 被演算データ $SOURCE_2$ または第 $r - 2$ 演算ステージ（図示略）の演算器の演算結果を保持する。ステージラッチ回路 $3 1 0_{r-1}$ は、第 $r - 2$ 演算ステージ（図示略）におけるステージラッチ回路の値を保持する。命令デコーダ $3 2 0_{r-11}$ は、ステージラッチ回路 $3 1 0_{r-1}$ からの命令 $INST_{r-11}$ をデコードする。

【0 1 8 3】

上記演算器 $1 2 0_{r-1}$ は、命令 $INST_{r-11}$ がデコードされたとき、ステージラッチ回路 $1 1 0_{r-1}$ の値を用いて、命令 $INST_{r-11}$ に対応する演算を行い演算結果をステージラッチ回路 $1 1 0_r$ へ出力する。また、命令デコーダ $3 2 0_{r-11}$ は、命令 $INST_{r-11}$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 $1 2 0_{r-1}$ に対してデコードする。演算器 $1 2 0_{r-1}$ は、スルー命令 SRC_1 がデコードされた場合、ステージラッチ回路 $1 1 0_{r-1}$ の値をスルーさせる。

【0 1 8 4】

命令デコーダ $3 2 0_{r-12}$ は、ステージラッチ回路 $3 1 0_{r-1}$ からの命令 $INST_{r-12}$ をデコードする。この演算器 $2 2 0_{r-1}$ は、命令 $INST_{r-12}$ がデコードされたとき、ステージラッチ回路 $2 1 0_{r-1}$ の値を用いて、命令 $INST_{r-12}$ に対応する演算を行い演算結果をステージラッチ回路 $2 1 0_r$ へ出力する。また、

命令デコーダ 320_{r-12} は、命令 $INST_{r-12}$ 以外の命令が入力された場合、これをスルー命令 SRC_2 に変換し、このスルー命令 SRC_2 を演算器 220_{r-1} に対してデコードする。演算器 220_{r-1} は、上記スルー命令 SRC_2 がデコードされた場合、ステージラッチ回路 210_{r-1} の値をスルーさせる。

【0185】

第 r 演算ステージにおいて、ステージラッチ回路 110_r は、第1被演算データ $SOURCE_1$ または第 $r-1$ 演算ステージにおける演算器 120_{r-1} の演算結果を保持する。ステージラッチ回路 210_r は、第2被演算データ $SOURCE_2$ または第 $r-1$ 演算ステージにおける演算器 220_{r-1} の演算結果を保持する。ステージラッチ回路 310_r は、ステージラッチ回路 310_{r-1} の値を保持する。

【0186】

命令デコーダ 320_r は、ステージラッチ回路 310_r からの命令 $INST_{rs}$ をデコードする。ここで命令 $INST_{rs}$ は、第 r 演算ステージ～第 s 演算ステージ（図18参照）における演算器 $120_r \sim$ 演算器 120_s に対して発行される命令であって、演算器 $120_r \sim$ 演算器 120_s における演算に関して相関関係を持たせるための命令である。ここで、第 r 演算ステージ～第 s 演算ステージの段数は、 $m(s-r+1)$ 段である。

【0187】

すなわち、命令 $INST_{rs}$ は、図17および図18に示した n 段（第1演算ステージ～第 n 演算ステージ）の演算ステージのうち、 m 段（第 r 演算ステージ～第 s 演算ステージ）の演算ステージ間において演算に関して相関関係を持たせるための命令である。演算器 120_r は、命令 $INST_{rs}$ がデコードされたとき、ステージラッチ回路 110_r の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_r の値（第2被演算データ $SOURCE_2$ ）を用いて、命令 $INST_{rs}$ に対応する演算を行い演算結果をステージラッチ回路 110_{r+1} へ出力する。

【0188】

また、命令デコーダ 320_r は、ステージラッチ回路 310_r からの命令 IN

ST_r をデコードする。ここで、命令 $INST_r$ は、他の演算ステージにおける演算器とは独立して演算器 120_r において演算を行わせるための命令である。演算器 120_r は、命令 $INST_r$ がデコードされたとき、ステージラッチ回路 110_r の値を用いて、命令 $INST_r$ に対応する演算を行い演算結果を出力する。

【0189】

さらに、命令デコーダ 320_r は、命令 $INST_{rs}$ および命令 $INST_r$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_r に対してデコードする。演算器 120_r は、スルー命令 SRC_1 がデコードされた場合、ステージラッチ回路 110_r の値、ステージラッチ回路 210_r の値のうち、ステージラッチ回路 110_r の値のみをスルーさせる。

【0190】

ここで、第 $r+1$ 演算ステージ～第 s 演算ステージ～第 $n-1$ 演算ステージ（図示略）のそれぞれの構成は、前述した図 5 に示した第 $x+1$ 演算ステージの構成と同様である。ただし、図 17 および図 18 に示した命令デコーダ 320_{r+1} ～命令デコーダ 320_s の機能は、後述するように、図 5 に示した命令デコーダ 320_{x+1} の機能とは異なる。

【0191】

第 $r+1$ 演算ステージにおいて、ステージラッチ回路 110_{r+1} は、第 1 被演算データ $SOURCE_1$ または第 r 演算ステージにおける演算器 120_r の演算結果を保持する。ステージラッチ回路 310_{r+1} は、ステージラッチ回路 310_r の値を保持する。命令デコーダ 320_{r+1} は、ステージラッチ回路 310_{r+1} からの上述した命令 $INST_{rs}$ をデコードする。演算器 120_{r+1} は、命令 $INST_{rs}$ がデコードされたとき、ステージラッチ回路 110_{r+1} の値（演算器 120_r の演算結果）を用いて、命令 $INST_{rs}$ に対応する演算を行い演算結果をマルチプレクサ 400_{r+1} へ出力する。

【0192】

また、命令デコーダ 320_{r+1} は、ステージラッチ回路 310_{r+1} からの命令

$INST_{r+1}$ をデコードする。ここで、命令 $INST_{r+1}$ は、他の演算ステージにおける演算器とは独立して演算器 120_{r+1} において演算を行わせるための命令である。演算器 120_{r+1} は、命令 $INST_{r+1}$ がデコードされたとき、ステージラッチ回路 110_{r+1} の値を用いて、命令 $INST_{r+1}$ に対応する演算を行い演算結果を出力する。バイパスライン B_{r+1} は、演算器 120_{r+1} に対して並列的に設けられており、ステージラッチ回路 110_{r+1} の値を、演算器 120_{r+1} を経由させることなく、スルーでマルチプレクサ 400_{r+1} に導く役目をしている。

【0193】

マルチプレクサ 400_{r+1} は、命令デコーダ 320_{r+1} により切り替え制御され、演算器 120_{r+1} の演算結果、ステージラッチ回路 110_{r+1} の値のうちいずれか一方を出力する。具体的には、マルチプレクサ 400_{r+1} は、命令デコーダ 320_{r+1} により命令 $INST_{rs}$ または命令 $INST_{r+1}$ がデコードされたとき演算器 120_{r+1} の演算結果を出力する。一方、マルチプレクサ 400_{r+1} は、命令 $INST_{rs}$ および命令 $INST_{r+1}$ 以外の命令が命令デコーダ 320_{r+1} に入力されたとき、ステージラッチ回路 110_{r+1} の値を出力する。ステージラッチ回路 110_{r+2} は、マルチプレクサ 400_{r+1} の切り替え状態に応じて、演算器 120_{r+1} の演算結果またはステージラッチ回路 110_{r+1} の値を保持する。ステージラッチ回路 110_{r+2} およびステージラッチ回路 310_{r+2} は、第 $r+1$ 演算ステージと第 $r+2$ 演算ステージ（図示略）との間にそれぞれ設けられており、マルチプレクサ 400_{r+1} の出力値、およびステージラッチ回路 310_{r+1} の値をそれぞれ保持する。以下、第 $r+2$ 演算ステージ～第 s 演算ステージまでは、上述した第 $r+1$ 演算ステージと同一構成とされている。

【0194】

すなわち、図 18 に示した第 s 演算ステージにおいて、ステージラッチ回路 110_s は、第 1 被演算データ $SOURCE_1$ または第 $s-1$ 演算ステージ（図示略）における演算器の演算結果を保持する。ステージラッチ回路 310_s は、ステージラッチ回路 310_{s-1} （図示略）の値を保持する。命令デコーダ 320_s は、ステージラッチ回路 310_s からの上述した命令 $INST_{rs}$ をデコードする

。演算器 120_s は、命令 $INST_{rs}$ がデコードされたとき、ステージラッチ回路 110_s の値を用いて、命令 $INST_{rs}$ に対応する演算を行い演算結果をマルチプレクサ 400_s へ出力する。

【0195】

また、命令デコーダ 320_s は、ステージラッチ回路 310_s からの命令 $INST_s$ をデコードする。ここで、命令 $INST_s$ は、他の演算ステージにおける演算器とは独立して演算器 120_s において演算を行わせるための命令である。演算器 120_s は、命令 $INST_s$ がデコードされたとき、ステージラッチ回路 110_s の値を用いて、命令 $INST_s$ に対応する演算を行い演算結果を出力する。バイパスライン B_s は、演算器 120_s に対して並列的に設けられており、ステージラッチ回路 110_s の値を、演算器 120_s を経由させることなく、スルーでマルチプレクサ 400_s に導く役目をしている。

【0196】

マルチプレクサ 400_s は、命令デコーダ 320_s により切り替え制御され、演算器 120_s の演算結果、ステージラッチ回路 110_s の値のうちいずれか一方を出力する。具体的には、マルチプレクサ 400_s は、命令デコーダ 320_s により命令 $INST_{rs}$ または命令 $INST_s$ がデコードされたとき演算器 120_s の演算結果を出力する。一方、マルチプレクサ 400_s は、命令 $INST_{rs}$ および命令 $INST_s$ 以外の命令が命令デコーダ 320_s に入力されたとき、ステージラッチ回路 110_s の値を出力する。ステージラッチ回路 110_{s+1} は、マルチプレクサ 400_s の切り替え状態に応じて、演算器 120_s の演算結果またはステージラッチ回路 110_s の値を保持する。ステージラッチ回路 110_{s+1} 、およびステージラッチ回路 310_{s+1} は、第 s 演算ステージと第 $s+1$ 演算ステージ（図示略）との間にそれぞれ設けられており、マルチプレクサ 400_s の出力値、およびステージラッチ回路 310_s の値をそれぞれ保持する。

【0197】

つぎに、上述した実施の形態 7 の動作について図 19～図 21 に示したフローチャートを参照しつつ説明する。はじめに、第 r 演算ステージの演算器 120_r ～第 s 演算ステージの演算器 120_s に対して命令 $INST_{rs}$ が発行された場合

の動作について説明する。図 19 に示したステップ SG 1 では、被演算データ用レジスタ 100、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第 1 被演算データ $SOURCE_1$ 、第 2 被演算データ $SOURCE_2$ および命令 $INST_{rs}$ がそれぞれ保持される。これにより、上記第 1 被演算データ $SOURCE_1$ 、第 2 被演算データ $SOURCE_2$ および命令 $INST_{rs}$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 に保持される。

【0198】

つぎのステップ SG 2 では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステージラッチ回路 310_1 に命令 $INST_{rs}$ が保持されているため、ステップ SG 5 へ進む。なお、ステージラッチ回路 310_1 に命令 $INST_{11}$ および命令 $INST_{12}$ が保持されている場合、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステップ SG 3 へ進む。この場合、ステップ SG 3 およびステップ SG 4 では、前述したステップ SC 3 およびステップ SC 4 (図 6 参照) の処理が行われる。

【0199】

この場合、ステップ SG 5 では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、命令 $INST_{rs}$ をスルー命令 SRC_1 およびスルー命令 SRC_2 にそれぞれ変換した後、ステップ SG 6 へ進む。ステップ SG 6 では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、演算器 120_1 および演算器 220_1 に対してスルー命令 SRC_1 およびスルー命令 SRC_2 をそれぞれデコードする。これにより、演算器 120_1 および演算器 220_1 は、ステージラッチ回路 110_1 の値 (第 1 被演算データ $SOURCE_1$) およびステージラッチ回路 210_1 の値 (第 2 被演算データ $SOURCE_2$) をスルーでそれぞれ通過させる。

【0200】

つぎのステップ SG 7 では、ステージラッチ回路 110_1 の値 (第 1 被演算データ $SOURCE_1$) およびステージラッチ回路 210_1 の値 (第 2 被演算データ $SOURCE_2$) がステージラッチ回路 110_2 およびステージラッチ回路 210_2 にそれぞれ保持される。このとき、ステージラッチ回路 310_1 の値 (命

令 $INST_{rs}$) がステージラッチ回路 310_2 に保持される。以後、第2演算ステージ～第 $r-2$ 演算ステージ (いずれも図示略) においては、第1演算ステージの場合と同様にして、スルー命令 SRC_1 およびスルー命令 SRC_2 がそれぞれデコードされることで、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ がそれぞれの演算ステージをスルーで順次通過する。また、第2演算ステージ～第 $r-2$ 演算ステージにおいては、命令 $INST_{rs}$ がステージラッチ回路に順次保持される。

【0201】

そして、第 $r-1$ 演算ステージにおいて、ステップSG8では、命令デコーダ 320_{r-11} および命令デコーダ 320_{r-12} は、ステージラッチ回路 310_{r-1} に命令 $INST_{rs}$ が保持されているため、ステップSG11へ進む。なお、ステージラッチ回路 310_{r-1} に命令 $INST_{r-11}$ および命令 $INST_{r-12}$ が保持されている場合、命令デコーダ 320_{r-11} および命令デコーダ 320_{r-12} は、ステップSG9へ進み、命令 $INST_{r-11}$ および命令 $INST_{r-12}$ を演算器 120_{r-1} および演算器 220_{r-1} に対してそれぞれデコードする。これにより、演算器 120_{r-1} および演算器 220_{r-1} はステージラッチ回路 110_{r-1} の値およびステージラッチ回路 210_{r-1} の値を用いて、命令 $INST_{r-11}$ および命令 $INST_{r-12}$ に応じた演算をそれぞれ行う。これにより、ステップSG10では、演算器 120_{r-1} および演算器 220_{r-1} の演算結果がステージラッチ回路 110_r およびステージラッチ回路 210_r に保持される。

【0202】

この場合、ステップSG11では、命令デコーダ 320_{r-11} および命令デコーダ 320_{r-12} は、命令 $INST_{rs}$ をスルー命令 SRC_1 およびスルー命令 SRC_2 にそれぞれ変換した後、ステップSG12へ進む。ステップSG12では、命令デコーダ 320_{r-11} および命令デコーダ 320_{r-12} は、演算器 120_{r-1} および演算器 220_{r-1} に対してスルー命令 SRC_1 およびスルー命令 SRC_2 をそれぞれデコードする。これにより、演算器 120_{r-1} および演算器 220_{r-1} は、ステージラッチ回路 110_{r-1} の値 (第1被演算データ $SOURCE_1$) およびステージラッチ回路 210_{r-1} の値 (第2被演算データ $SOURCE_2$) をス

ルーでそれぞれ通過させる。

【0203】

つぎのステップSG13では、ステージラッチ回路 110_{r-1} の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_{r-1} の値（第2被演算データ $SOURCE_2$ ）がステージラッチ回路 110_r およびステージラッチ回路 210_r にそれぞれ保持される。このとき、ステージラッチ回路 310_{r-1} の値（命令 $INST_{rs}$ ）がステージラッチ回路 310_r に保持される。

【0204】

つぎの第 r 演算ステージにおいて、ステップSG14（図20参照）では、命令デコーダ 320_r は、ステージラッチ回路 310_r に命令 $INST_{rs}$ が保持されているため、ステップSG15へ進む。なお、ステージラッチ回路 310_r に命令 $INST_{rs}$ および命令 $INST_r$ 以外の命令が保持されている場合、命令デコーダ 320_r は、ステップSG17へ進み、命令 $INST_{rs}$ および $INST_r$ 以外の命令をスルー命令 SRC_1 に変換した後、ステップSG18へ進む。ステップSG18では、命令デコーダ 320_r は、演算器 120_r に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_r は、ステージラッチ回路 110_r の値をスルーで通過させる。つぎのステップSG19では、ステージラッチ回路 110_r の値がステージラッチ回路 110_{r+1} に保持される。

【0205】

この場合、ステップSG15では、命令デコーダ 320_r は、命令 $INST_{rs}$ を演算器 120_r に対してデコードする。これにより、演算器 120_r はステージラッチ回路 110_r の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_r の値（第2被演算データ $SOURCE_2$ ）を用いて、命令 $INST_{rs}$ に応じた演算を行う。これにより、ステップSG16では、演算器 120_r の演算結果がステージラッチ回路 110_{r+1} に保持される。このとき、ステージラッチ回路 310_r の値（命令 $INST_{rs}$ ）がステージラッチ回路 310_{r+1} に保持される。

【0206】

つぎの第 $r+1$ 演算ステージにおいて、ステップSG20では、命令デコーダ

320_{r+1} は、ステージラッチ回路 310_{r+1} に命令 $INST_{rs}$ が保持されているため、ステップ SG 2 1 へ進む。なお、ステージラッチ回路 310_{r+1} に命令 $INST_{rs}$ および命令 $INST_{r+1}$ 以外の命令が保持されている場合、命令デコーダ 320_{r+1} は、ステップ SG 2 3 へ進み、マルチプレクサ 400_{r+1} にステージラッチ回路 110_{r+1} を選択させる。これにより、ステップ SG 2 4 では、演算器 120_{r+1} が動作せず、ステップ SG 2 5 では、ステージラッチ回路 110_{r+1} の値が、バイパスライン B_{r+1} およびマルチプレクサ 400_{r+1} を介してステージラッチ回路 110_{r+2} に保持される。

【0207】

この場合、ステップ SG 2 1 では、命令デコーダ 320_{r+1} は、マルチプレクサ 400_{r+1} に演算器 120_{r+1} を選択させる。また、命令デコーダ 320_{r+1} は、ステージラッチ回路 310_{r+1} の値を解読した後、命令 $INST_{rs}$ を演算器 120_{r+1} に対してデコードする。これにより、演算器 120_{r+1} は、ステージラッチ回路 110_{r+1} の値を用いて、命令 $INST_{rs}$ に応じた演算を行う。つぎのステップ SG 2 2 では、演算器 120_{r+1} の演算結果がステージラッチ回路 110_{r+2} に保持される。以後、第 $r+2$ 演算ステージ～第 $s-1$ 演算ステージ（いずれも図示略）においては、第 $r+1$ 演算ステージの場合と同様にして、命令 $INST_{rs}$ がデコードされるとともに、マルチプレクサにより演算器が選択されることで、それぞれの演算ステージにおいて命令 $INST_{rs}$ に基づく演算が行われる。また、第 $r+2$ 演算ステージ～第 $s-1$ 演算ステージにおいては、命令 $INST_{rs}$ がステージラッチ回路に順次保持される。

【0208】

そして、図 1 8 に示した第 s 演算ステージにおいて、ステップ SG 2 6（図 2 1 参照）では、命令デコーダ 320_s は、ステージラッチ回路 310_s に命令 $INST_{rs}$ が保持されているため、ステップ SG 2 7 へ進む。なお、ステージラッチ回路 310_s に命令 $INST_{rs}$ および命令 $INST_s$ 以外の命令が保持されている場合、命令デコーダ 320_s は、ステップ SG 2 9 へ進み、マルチプレクサ 400_s にステージラッチ回路 110_s を選択させる。これにより、ステップ SG 3 0 では、演算器 120_s が動作せず、ステップ SG 3 1 では、ステージラッ

チ回路 110_s の値が、バイパスライン B_s およびマルチプレクサ 400_s を介してステージラッチ回路 110_{s+1} に保持される。

【0209】

この場合、ステップ SG 27 では、命令デコーダ 320_s は、マルチプレクサ 400_s に演算器 120_s を選択させる。また、命令デコーダ 320_s は、命令 $INST_{rs}$ を演算器 120_s に対してデコードする。これにより、演算器 120_s は、ステージラッチ回路 110_s の値（第 $s-1$ 演算ステージにおける演算器の演算結果）を用いて、命令 $INST_{rs}$ に応じた演算を行う。つぎのステップ SG 28 では、演算器 120_s の演算結果がステージラッチ回路 110_{s+1} に保持される。ここで、演算器 120_s の演算結果は、演算器 $120_r \sim$ 演算器 120_s におけるそれぞれの演算に基づくものである。以後、第 $s+2$ 演算ステージ～第 $n-1$ 演算ステージ（いずれも図示略）においては、マルチプレクサによりステージラッチ回路が選択されることで、演算器 120_s の演算結果がそれぞれの演算ステージにおいて順次スルーされる。また、第 $s+2$ 演算ステージ～第 $n-1$ 演算ステージにおいては、命令 $INST_{rs}$ がステージラッチ回路に順次保持される。

【0210】

そして、第 n 演算ステージにおいて、ステップ SG 32 では、命令デコーダ 320_n は、ステージラッチ回路 310_n に命令 $INST_{rs}$ が保持されているため、ステップ SG 35 へ進む。なお、ステージラッチ回路 310_n に命令 $INST_n$ が保持されている場合、命令デコーダ 320_n は、ステップ SG 33 へ進み、マルチプレクサ 400_n に演算器 120_n を選択させるとともに、命令 $INST_n$ を演算器 120_n に対してデコードする。これにより、演算器 120_n は、ステージラッチ回路 110_n の値を用いて、命令 $INST_n$ に応じた演算を行う。つぎのステップ SG 34 では、演算器 120_n の演算結果がステージラッチ回路 110_{n+1} に保持される。

【0211】

この場合、ステップ SG 35 では、命令デコーダ 320_n は、マルチプレクサ 400_n にステージラッチ回路 110_n を選択させる。これにより、ステップ S

G 3 6では、演算器 120_n が動作せず、ステップSG 3 7では、ステージラッチ回路 110_n の値（演算器 120_s の演算結果）が、バイパスライン B_n およびマルチプレクサ 400_n を介してステージラッチ回路 110_{n+1} に保持される。つぎのステップSG 3 8では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_s の演算結果が保持される。

【0 2 1 2】

つぎに、図1 7および図1 8に示した m 段（第 r 演算ステージ～第 s 演算ステージ）の演算ステージの中に存在する一つの演算ステージ（以下、第 x 演算ステージと称する）における演算器に対して、他の演算器とは独立に演算を実行させるための命令（以下、命令 $INST_x$ と称する）が発行された場合の動作について説明する。以下の説明においては、一例として、上記第 x 演算ステージおよび命令 $INST_x$ を、図1 7に示した第 $r+1$ 演算ステージおよび命令 $INST_{r+1}$ とした場合について説明する。図1 9に示したステップSG 1では、被演算データ用レジスタ 100 、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第1被演算データ $SOURCE_1$ 、第2被演算データ $SOURCE_2$ および命令 $INST_{r+1}$ がそれぞれ保持される。これにより、上記第1被演算データ $SOURCE_1$ 、第2被演算データ $SOURCE_2$ および命令 $INST_{r+1}$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 に保持される。

【0 2 1 3】

つぎのステップSG 2では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステージラッチ回路 310_1 に命令 $INST_{r+1}$ が保持されているため、ステップSG 5へ進む。ステップSG 5～ステップSG 7においては、前述と同様にして、スルー命令 SRC_1 およびスルー命令 SRC_2 がそれぞれデコードされることで、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ がステージラッチ回路 110_2 およびステージラッチ回路 210_2 にそれぞれ保持される。また、ステージラッチ回路 310_2 には、命令 $INST_{r+1}$ が保持される。

【0 2 1 4】

以後、第 2 演算ステージ～第 $r - 2$ 演算ステージ（いずれも図示略）においては、第 1 演算ステージの場合と同様にして、スルー命令 SRC_1 およびスルー命令 SRC_2 がそれぞれデコードされることで、第 1 被演算データ $SOURCE_1$ および第 2 被演算データ $SOURCE_2$ がそれぞれの演算ステージをスルーで順次通過する。また、第 2 演算ステージ～第 $r - 2$ 演算ステージにおいては、命令 $INST_{r+1}$ がステージラッチ回路に順次保持される。

【0 2 1 5】

そして、第 $r - 1$ 演算ステージにおいて、ステップ SG 8 では、命令デコーダ 320_{r-11} および命令デコーダ 320_{r-12} は、ステージラッチ回路 310_{r-1} に命令 $INST_{r+1}$ が保持されているため、ステップ SG 11 へ進む。ステップ SG 11～ステップ SG 13 においては、前述と同様にして、スルー命令 SRC_1 およびスルー命令 SRC_2 がそれぞれデコードされることで、第 1 被演算データ $SOURCE_1$ および第 2 被演算データ $SOURCE_2$ がステージラッチ回路 110_r およびステージラッチ回路 210_r にそれぞれ保持される。また、ステージラッチ回路 310_r には、命令 $INST_{r+1}$ が保持される。

【0 2 1 6】

つぎの第 r 演算ステージにおいて、ステップ SG 14（図 20 参照）では、命令デコーダ 320_r は、ステージラッチ回路 310_r に命令 $INST_{r+1}$ が保持されているため、ステップ SG 17 へ進む。ステップ SG 17～ステップ SG 19 においては、前述と同様にして、スルー命令 SRC_1 がデコードされることにより、第 1 被演算データ $SOURCE_1$ がステージラッチ回路 110_{r+1} に保持される。このとき、ステージラッチ回路 310_{r+1} には、命令 $INST_{r+1}$ が保持される。

【0 2 1 7】

つぎの第 $r + 1$ 演算ステージにおいて、ステップ SG 20 では、命令デコーダ 320_{r+1} は、ステージラッチ回路 310_{r+1} に命令 $INST_{r+1}$ が保持されているため、ステップ SG 21 へ進む。ステップ SG 21 では、命令デコーダ 320_{r+1} は、マルチプレクサ 400_{r+1} に演算器 120_{r+1} を選択させる。また、命令デコーダ 320_{r+1} は、ステージラッチ回路 310_{r+1} の値を解読した後、

命令 $INST_{r+1}$ を演算器 120_{r+1} に対してデコードする。これにより、演算器 120_{r+1} は、ステージラッチ回路 110_{r+1} の値を用いて、命令 $INST_{r+1}$ に応じた演算を行う。つぎのステップ SG 2 2 では、演算器 120_{r+1} の演算結果がステージラッチ回路 110_{r+2} に保持される。以後、第 $r+2$ 演算ステージ～第 $s-1$ 演算ステージ（いずれも図示略）においては、マルチプレクサによりステージラッチ回路が選択されることで、それぞれの演算ステージにおいて第 $r+1$ 演算ステージにおける演算器 120_{r+1} の演算結果が順次スルーされる。また、第 $r+2$ 演算ステージ～第 $s-1$ 演算ステージにおいては、命令 $INST_{r+1}$ がステージラッチ回路に順次保持される。

【0 2 1 8】

そして、第 s 演算ステージにおいて、ステップ SG 2 6（図 2 1 参照）では、命令デコーダ 320_s は、ステージラッチ回路 310_s に命令 $INST_{r+1}$ が保持されているため、ステップ SG 2 9 へ進む。ステップ SG 2 9～ステップ SG 3 1 においては、前述と同様にしてマルチプレクサ 400_s によりステージラッチ回路 110_s が選択されることで、第 $r+1$ 演算ステージにおける演算器 120_{r+1} の演算結果がステージラッチ回路 110_{s+1} に保持される。以後、第 $s+1$ 演算ステージ～第 $n-1$ 演算ステージ（いずれも図示略）においては、マルチプレクサによりステージラッチ回路が選択されることで、第 $r+1$ 演算ステージにおける演算器 120_{r+1} の演算結果がそれぞれのステージにおいて順次スルーされる。また、第 $s+1$ 演算ステージ～第 $n-1$ 演算ステージにおいては、命令 $INST_{r+1}$ がステージラッチ回路に順次保持される。

【0 2 1 9】

そして、第 n 演算ステージにおいて、ステップ SG 3 2 では、命令デコーダ 320_n は、ステージラッチ回路 310_n に命令 $INST_{r+1}$ が保持されているため、ステップ SG 3 5 へ進む。ステップ SG 3 5～ステップ SG 3 8 においては、前述と同様にして、マルチプレクサ 400_n によりステージラッチ回路 110_n が選択されることで、第 $r+1$ 演算ステージにおける演算器 120_{r+1} の演算結果が、演算結果用レジスタ 500 に保持される。

【0 2 2 0】

以上説明したように、実施の形態 7 によれば、第 r 演算ステージ～第 s 演算ステージにおける演算実行時に、第 1 演算ステージ～第 $r - 1$ 演算ステージにおいてスルー命令 SRC_1 およびスルー命令 SRC_2 をそれぞれデコードすることで第 1 被演算データ $SOURCE_1$ および第 2 被演算データ $SOURCE_2$ をスルーさせるようにしたので、第 1 演算ステージ～第 $r - 1$ 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減する。また、実施の形態 7 によれば、第 r 演算ステージ～第 s 演算ステージのうちいずれか一つの演算ステージにおける演算器の演算を、他の演算器とは独立させて行うことができる。

【0221】

なお、上述した実施の形態 7 においては、図 17 および図 18 に示した m 段の第 r 演算ステージ～第 s 演算ステージのうちいずれか一つの演算ステージの演算器で演算を単独実行するように構成した例について説明したが、特定の一つの演算ステージにおける演算器のみで演算を実行するようにしてもよい。また、実施の形態 7 においては、 m 段の第 r 演算ステージ～第 s 演算ステージのうち、第 x 演算ステージ～第 $x + p$ ($p \leq s - r$) 演算ステージのそれぞれの演算器で演算を単独実行するようにしてもよい。

【0222】

(実施の形態 8)

さて、前述した実施の形態 4 において、図 9 に示した第 1 演算ステージ～第 n 演算ステージでは、互いに相関関係がない命令に基づいて演算をそれぞれ行う場合について説明したが、相関関係がある命令に基づいて演算を行う複数の演算器を備えるようにしてもよい。以下においては、この場合を実施の形態 8 として詳述する。

【0223】

図 22 および図 23 は、本発明にかかる実施の形態 8 の構成を示すブロック図である。この図において、図 9 の各部にそれぞれ対応する部分には同一の符号を付け、その説明を省略する。図 22 および図 23 においては、図 9 に示した第 $x - 1$ 演算ステージ～第 $x + 1$ 演算ステージに代えて、第 $r - 1$ 演算ステージ、第

r 演算ステージ、第 $r + 1$ 演算ステージおよび第 s 演算ステージ（ただし、 $1 < r < s < n$ ）がそれぞれ図示されている。図 2 2 および図 2 3 に示した第 $r - 1$ 演算ステージ、第 r 演算ステージおよび第 $r + 1$ 演算ステージ（第 s 演算ステージ）のそれぞれの構成は、図 9 に示した第 $x - 1$ 演算ステージ、第 x 演算ステージおよび第 $x + 1$ 演算ステージのそれぞれの構成と同様である。ただし、図 2 2 および図 2 3 に示した第 $r - 1$ 演算ステージ、第 r 演算ステージおよび第 $r + 1$ 演算ステージ（第 s 演算ステージ）における命令デコーダの機能が、図 9 に示した命令デコーダの機能と異なる。

【0 2 2 4】

すなわち、図 2 2 に示した第 $r - 1$ 演算ステージにおいて、ステージラッチ回路 $1 1 0_{r-1}$ は、第 1 被演算データ $SOURCE_1$ または第 $r - 2$ 演算ステージ（図示略）の演算器の演算結果を保持する。ステージラッチ回路 $2 1 0_{r-1}$ は、第 2 被演算データ $SOURCE_2$ または第 $r - 2$ 演算ステージ（図示略）の演算器の演算結果を保持する。ステージラッチ回路 $3 1 0_{r-1}$ は、第 $r - 2$ 演算ステージ（図示略）におけるステージラッチ回路の値を保持する。命令デコーダ $3 2 0_{r-11}$ は、ステージラッチ回路 $3 1 0_{r-1}$ からの命令 $INST_{r-11}$ をデコードする。演算器 $1 2 0_{r-1}$ は、命令 $INST_{r-11}$ がデコードされたとき、ステージラッチ回路 $1 1 0_{r-1}$ の値を用いて、命令 $INST_{r-11}$ に対応する演算を行う。バイパスライン B_{r-11} は、演算器 $1 2 0_{r-1}$ を経由させることなく、ステージラッチ回路 $1 1 0_{r-1}$ の値をマルチプレクサ $7 0 0_{r-1}$ に導く役目をしている。

【0 2 2 5】

マルチプレクサ $7 0 0_{r-1}$ は、命令デコーダ $3 2 0_{r-11}$ により切り替え制御され、命令デコーダ $3 2 0_{r-11}$ により命令 $INST_{r-11}$ がデコードされたとき演算器 $1 2 0_{r-1}$ の演算結果を出力する。一方、マルチプレクサ $7 0 0_{r-1}$ は、命令 $INST_{r-11}$ 以外の命令が命令デコーダ $3 2 0_{r-11}$ に入力されたとき、ステージラッチ回路 $1 1 0_{r-1}$ の値を出力する。

【0 2 2 6】

命令デコーダ $3 2 0_{r-12}$ は、ステージラッチ回路 $3 1 0_{r-1}$ からの命令 $INST_{12}$ をデコードする。演算器 $2 2 0_{r-1}$ は、命令 $INST_{r-12}$ がデコードされた

とき、ステージラッチ回路 210_{r-1} の値を用いて、命令 $INST_{r-12}$ に対応する演算を行う。バイパスライン B_{r-12} は、演算器 220_{r-1} を経由させることなく、ステージラッチ回路 210_{r-1} の値をマルチプレクサ 800_{r-1} に導く役目をしている。マルチプレクサ 800_{r-1} は、命令デコーダ 320_{r-12} により命令 $INST_{r-12}$ がデコードされたとき演算器 220_{r-1} の演算結果を出力し、命令 $INST_{r-12}$ 以外の命令が命令デコーダ 320_{r-12} に入力されたとき、ステージラッチ回路 210_{r-1} の値を出力する。

【0227】

つぎの第 r 演算ステージにおいて、ステージラッチ回路 110_r 、ステージラッチ回路 210_r およびステージラッチ回路 310_r は、マルチプレクサ 700_{r-1} の出力値、マルチプレクサ 800_{r-1} の出力値およびステージラッチ回路 310_{r-1} の値をそれぞれ保持する。命令デコーダ 320_r は、ステージラッチ回路 310_r からの命令 $INST_{rs}$ をデコードする。ここで命令 $INST_{rs}$ は、前述した実施の形態 7 と同様にして、第 r 演算ステージ～第 s 演算ステージ（図 23 参照）における演算器 $120_r \sim$ 演算器 120_s に対して発行される命令であって、演算器 $120_r \sim$ 演算器 120_s における演算に関して相関関係を持たせるための命令である。ここで、第 r 演算ステージ～第 s 演算ステージの段数は、 $m(s-r+1)$ 段である。

【0228】

すなわち、命令 $INST_{rs}$ は、 n 段（第 1 演算ステージ～第 n 演算ステージ）の演算ステージのうち、 m 段（第 r 演算ステージ～第 s 演算ステージ）の演算ステージ間において演算に関して相関関係を持たせるための命令である。演算器 120_r は、命令 $INST_{rs}$ がデコードされたとき、ステージラッチ回路 110_r の値（第 1 被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_r の値（第 2 被演算データ $SOURCE_2$ ）を用いて、命令 $INST_{rs}$ に対応する演算を行い演算結果をステージラッチ回路 110_{r+1} へ出力する。

【0229】

また、命令デコーダ 320_r は、ステージラッチ回路 310_r からの命令 $INST_r$ をデコードする。ここで、命令 $INST_r$ は、他の演算ステージにおける

演算器とは独立して演算器 120_r において演算を行わせるための命令である。
 演算器 120_r は、命令 $INST_r$ がデコードされたとき、ステージラッチ回路 110_r の値を用いて、命令 $INST_r$ に対応する演算を行い演算結果を出力する。

【0230】

さらに、命令デコーダ 320_r は、命令 $INST_{rs}$ および命令 $INST_r$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_r に対してデコードする。演算器 120_r は、ステージラッチ回路 110_r の値、ステージラッチ回路 210_r の値のうち、ステージラッチ回路 110_r の値のみをスルーさせる。

【0231】

ここで、第 $r+1$ 演算ステージ～第 s 演算ステージのそれぞれの構成は、前述した図9に示した第 $x+1$ 演算ステージの構成と同様である。ただし、図22および図23に示した命令デコーダ 320_{r+1} ～命令デコーダ 320_s の機能は、後述するように、図9に示した命令デコーダ 320_{x+1} の機能とは異なる。

【0232】

第 $r+1$ 演算ステージにおいて、ステージラッチ回路 110_{r+1} は、ステージラッチ回路 110_r の値または演算器 120_r の演算結果を保持する。ステージラッチ回路 310_{r+1} は、ステージラッチ回路 310_r の値を保持する。命令デコーダ 320_{r+1} は、ステージラッチ回路 310_{r+1} からの命令 $INST_{rs}$ をデコードする。演算器 120_{r+1} は、命令 $INST_{rs}$ がデコードされたとき、ステージラッチ回路 110_{r+1} の値を用いて、命令 $INST_{rs}$ に対応する演算を行い演算結果をステージラッチ回路 110_{r+2} へ出力する。

【0233】

また、命令デコーダ 320_{r+1} は、ステージラッチ回路 310_{r+1} からの命令 $INST_{r+1}$ をデコードする。ここで、命令 $INST_{r+1}$ は、他の演算ステージにおける演算器とは独立して演算器 120_{r+1} において演算を行わせるための命令である。演算器 120_{r+1} は、命令 $INST_{r+1}$ がデコードされたとき、ステージラッチ回路 110_{r+1} の値を用いて、命令 $INST_{r+1}$ に対応する演算を行

い演算結果を出力する。

【0 2 3 4】

さらに、命令デコーダ 320_{r+1} は、命令 $INST_{rs}$ および命令 $INST_{r+1}$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_{r+1} に対してデコードする。演算器 120_{r+1} は、ステージラッチ回路 110_{r+1} の値をスルーさせる。以下、第 $r+2$ 演算ステージ（図示略）～第 s 演算ステージまでは、上述した第 $r+1$ 演算ステージと同一構成とされている。

【0 2 3 5】

すなわち、図 2 3 に示した第 s 演算ステージにおいて、ステージラッチ回路 110_s は、第 $s-1$ 演算ステージ（図示略）におけるステージラッチ回路の値または演算器の演算結果を保持する。命令デコーダ 320_s は、ステージラッチ回路 310_s からの命令 $INST_{rs}$ をデコードする。演算器 120_s は、命令 $INST_{rs}$ がデコードされたとき、ステージラッチ回路 110_s の値を用いて、命令 $INST_{rs}$ に対応する演算を行い演算結果をステージラッチ回路 110_{s+1} へ出力する。

【0 2 3 6】

また、命令デコーダ 320_s は、ステージラッチ回路 310_s からの命令 $INST_s$ をデコードする。ここで、命令 $INST_s$ は、他の演算ステージにおける演算器とは独立して演算器 120_s において演算を行わせるための命令である。演算器 120_s は、命令 $INST_s$ がデコードされたとき、ステージラッチ回路 110_s の値を用いて、命令 $INST_s$ に対応する演算を行い演算結果を出力する。

【0 2 3 7】

さらに、命令デコーダ 320_s は、命令 $INST_{rs}$ および命令 $INST_s$ 以外の命令が入力された場合、これをスルー命令 SRC_1 に変換し、このスルー命令 SRC_1 を演算器 120_s に対してデコードする。演算器 120_s は、ステージラッチ回路 110_s の値をスルーさせる。ステージラッチ回路 110_{s+1} およびステージラッチ回路 310_{s+1} は、演算器 120_s の出力値およびステージラッ

チ回路 3 1 0_s の値をそれぞれ保持する。

【0 2 3 8】

つぎに、上述した実施の形態 8 の動作について図 2 4 ～図 2 6 に示したフローチャートを参照しつつ説明する。はじめに、第 r 演算ステージの演算器 1 2 0_r ～第 s 演算ステージの演算器 1 2 0_s に対して命令 INST_{rs} が発行された場合の動作について説明する。図 2 4 に示したステップ SH 1 では、被演算データ用レジスタ 1 0 0、被演算データ用レジスタ 2 0 0 および命令用レジスタ 3 0 0 に、第 1 被演算データ SOURCE₁、第 2 被演算データ SOURCE₂ および命令 INST_{rs} がそれぞれ保持される。これにより、上記第 1 被演算データ SOURCE₁、第 2 被演算データ SOURCE₂ および命令 INST_{rs} がステージラッチ回路 1 1 0₁、ステージラッチ回路 2 1 0₁ およびステージラッチ回路 3 1 0₁ にそれぞれ保持される。

【0 2 3 9】

つぎのステップ SH 2 では、命令デコーダ 3 2 0₁₁ および命令デコーダ 3 2 0₁₂ は、ステージラッチ回路 3 1 0₁ に命令 INST_{rs} が保持されているため、ステップ SH 5 へ進む。なお、ステージラッチ回路 3 1 0₁ に命令 INST₁₁ および命令 INST₁₂ が保持されている場合、命令デコーダ 3 2 0₁₁ および命令デコーダ 3 2 0₁₂ は、ステップ SH 3 へ進む。この場合、ステップ SH 3 およびステップ SH 4 では、前述したステップ SD 3 およびステップ SD 4（図 1 0 参照）の処理が行われる。

【0 2 4 0】

この場合、ステップ SH 5 では、命令デコーダ 3 2 0₁₁ および命令デコーダ 3 2 0₁₂ は、マルチプレクサ 7 0 0₁ およびマルチプレクサ 8 0 0₁ にステージラッチ回路 1 1 0₁ およびステージラッチ回路 2 1 0₁ を選択させる。ステップ SH 6 では、演算器 1 2 0₁ および演算器 2 2 0₁ が動作せず、ステップ SH 7 では、ステージラッチ回路 1 1 0₁ の値（第 1 被演算データ SOURCE₁）が、バイパスライン B₁₁ およびマルチプレクサ 7 0 0₁ を介してステージラッチ回路 1 1 0₂ に保持される。同様に、ステージラッチ回路 2 1 0₁ の値（第 2 被演算データ SOURCE₂）が、バイパスライン B₁₂ およびマルチプレクサ 8 0

0_1 を介してステージラッチ回路 $2\ 1\ 0_2$ に保持される。また、ステージラッチ回路 $3\ 1\ 0_2$ には、命令 $INST_{rs}$ が保持される。

【0 2 4 1】

以後、第 2 演算ステージ～第 $r - 2$ 演算ステージ（いずれも図示略）においては、第 1 演算ステージの場合と同様にして、マルチプレクサによりステージラッチ回路が選択されることで、第 1 被演算データ $SOURCE_1$ および第 2 被演算データ $SOURCE_2$ がそれぞれの演算ステージをスルーで順次通過する。また、第 2 演算ステージ～第 $r - 2$ 演算ステージにおいては、命令 $INST_{rs}$ がステージラッチ回路に順次保持される。

【0 2 4 2】

そして、第 $r - 1$ 演算ステージにおいて、ステップ SH 8 では、命令デコーダ $3\ 2\ 0_{r-11}$ および命令デコーダ $3\ 2\ 0_{r-12}$ は、ステージラッチ回路 $3\ 1\ 0_{r-1}$ に命令 $INST_{rs}$ が保持されているため、ステップ SH 11 へ進む。なお、ステージラッチ回路 $3\ 1\ 0_{r-1}$ に命令 $INST_{r-11}$ および命令 $INST_{r-12}$ が保持されている場合、命令デコーダ $3\ 2\ 0_{r-11}$ および命令デコーダ $3\ 2\ 0_{r-12}$ は、ステップ SH 9 へ進み、マルチプレクサ $7\ 0\ 0_{r-1}$ およびマルチプレクサ $8\ 0\ 0_{r-1}$ に演算器 $1\ 2\ 0_{r-1}$ および演算器 $2\ 2\ 0_{r-1}$ を選択させる。

【0 2 4 3】

また、命令デコーダ $3\ 2\ 0_{r-11}$ および命令デコーダ $3\ 2\ 0_{r-12}$ は、命令 $INST_{r-11}$ および命令 $INST_{r-12}$ を演算器 $1\ 2\ 0_{r-1}$ および演算器 $2\ 2\ 0_{r-1}$ に対してそれぞれデコードする。これにより、演算器 $1\ 2\ 0_{r-1}$ および演算器 $2\ 2\ 0_{r-1}$ は、ステージラッチ回路 $1\ 1\ 0_{r-1}$ の値およびステージラッチ回路 $2\ 1\ 0_{r-1}$ の値を用いて、命令 $INST_{r-11}$ および命令 $INST_{r-12}$ に応じた演算を行う。つぎのステップ SH 10 では、演算器 $1\ 2\ 0_{r-1}$ および演算器 $2\ 2\ 0_{r-1}$ の演算結果がステージラッチ回路 $1\ 1\ 0_r$ およびステージラッチ回路 $2\ 1\ 0_r$ にそれぞれ保持される。

【0 2 4 4】

この場合、ステップ SH 11 では、命令デコーダ $3\ 2\ 0_{r-11}$ および命令デコーダ $3\ 2\ 0_{r-12}$ は、マルチプレクサ $7\ 0\ 0_{r-1}$ およびマルチプレクサ $8\ 0\ 0_{r-1}$ に

ステージラッチ回路 110_{r-1} およびステージラッチ回路 210_{r-1} を選択させる。ステップ SH12 では、演算器 120_{r-1} および演算器 220_{r-1} が動作せず、ステップ SH13 では、ステージラッチ回路 110_{r-1} の値（第1被演算データ $SOURCE_1$ ）が、バイパスライン B_{r-11} およびマルチプレクサ 700_{r-1} を介してステージラッチ回路 110_r に保持される。同様にして、ステージラッチ回路 210_{r-1} の値（第2被演算データ $SOURCE_2$ ）が、バイパスライン B_{r-12} およびマルチプレクサ 800_{r-1} を介してステージラッチ回路 210_r に保持される。

【0245】

つぎの第 r 演算ステージにおいて、ステップ SH14（図25参照）では、命令デコーダ 320_r は、ステージラッチ回路 310_r に命令 $INST_{rs}$ が保持されているため、ステップ SH15 へ進む。なお、ステージラッチ回路 310_r に命令 $INST_{rs}$ および命令 $INST_r$ 以外の命令が保持されている場合、命令デコーダ 320_r は、ステップ SH17 へ進み、命令 $INST_{rs}$ および $INST_r$ 以外の命令をスルー命令 SRC_1 に変換した後、ステップ SH18 へ進む。ステップ SH18 では、命令デコーダ 320_r は、演算器 120_r に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_r は、ステージラッチ回路 110_r の値をスルーで通過させる。つぎのステップ SH19 では、ステージラッチ回路 110_r の値がステージラッチ回路 110_{r+1} に保持される。

【0246】

この場合、ステップ SH15 では、命令デコーダ 320_r は、命令 $INST_{rs}$ を演算器 120_r に対してデコードする。これにより、演算器 120_r はステージラッチ回路 110_r の値（第1被演算データ $SOURCE_1$ ）およびステージラッチ回路 210_r の値（第2被演算データ $SOURCE_2$ ）を用いて、命令 $INST_{rs}$ に応じた演算を行う。これにより、ステップ SH16 では、演算器 120_r の演算結果がステージラッチ回路 110_{r+1} に保持される。このとき、ステージラッチ回路 310_r の値（命令 $INST_{rs}$ ）がステージラッチ回路 310_{r+1} に保持される。

【0247】

つぎの第 $r+1$ 演算ステージにおいて、ステップ SH20 では、命令デコーダ 320_{r+1} は、ステージラッチ回路 310_{r+1} に命令 $INST_{rs}$ が保持されているため、ステップ SH21 へ進む。なお、ステージラッチ回路 310_{r+1} に命令 $INST_{rs}$ および命令 $INST_{r+1}$ 以外の命令が保持されている場合、命令デコーダ 320_{r+1} は、ステップ SH23 へ進み、命令 $INST_{rs}$ および $INST_{r+1}$ 以外の命令をスルー命令 SRC_1 に変換した後、ステップ SH24 へ進む。ステップ SH24 では、命令デコーダ 320_{r+1} は、演算器 120_{r+1} に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_{r+1} は、ステージラッチ回路 110_{r+1} の値をスルーで通過させる。つぎのステップ SH25 では、ステージラッチ回路 110_{r+1} の値がステージラッチ回路 110_{r+2} に保持される。

【0248】

この場合、ステップ SH21 では、命令デコーダ 320_{r+1} は、命令 $INST_{rs}$ を演算器 120_{r+1} に対してデコードする。これにより、演算器 120_{r+1} はステージラッチ回路 110_{r+1} の値（第 r 演算ステージの演算器 120_r の演算結果）を用いて、命令 $INST_{rs}$ に応じた演算を行う。これにより、ステップ SH22 では、演算器 120_{r+1} の演算結果がステージラッチ回路 110_{r+2} に保持される。このとき、ステージラッチ回路 310_{r+1} の値（命令 $INST_{rs}$ ）がステージラッチ回路 310_{r+2} に保持される。以後、第 $r+2$ 演算ステージ～第 $s-1$ 演算ステージ（いずれも図示略）においては、第 $r+1$ 演算ステージと同様にして、命令 $INST_{rs}$ がデコードされることで、それぞれの演算ステージにおいて命令 $INST_{rs}$ に基づく演算が行われる。また、第 $r+2$ 演算ステージ～第 $s-1$ 演算ステージにおいては、命令 $INST_{rs}$ がステージラッチ回路に順次保持される。

【0249】

そして、図 23 に示した第 s 演算ステージにおいて、ステップ SH26（図 26 参照）では、命令デコーダ 320_s は、ステージラッチ回路 310_s に命令 $INST_{rs}$ が保持されているため、ステップ SH27 へ進む。なお、ステージラッチ回路 310_s に命令 $INST_{rs}$ および命令 $INST_s$ 以外の命令が保持されて

いる場合、命令デコーダ 320_s は、ステップ SH29 へ進み、命令 $INST_{rs}$ および命令 $INST_s$ 以外の命令をスルー命令 SRC_1 に変換した後、ステップ SH30 へ進む。ステップ SH30 では、演算器 120_s に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_s は、ステージラッチ回路 110_s の値をスルーで通過させる。つぎのステップ SH31 では、ステージラッチ回路 110_s の値がステージラッチ回路 110_{s+1} に保持される。

【0250】

この場合、ステップ SH27 では、命令デコーダ 320_s は、命令 $INST_{rs}$ を演算器 120_s に対してデコードする。これにより、演算器 120_s はステージラッチ回路 110_s の値（第 $s-1$ 演算ステージにおける演算器の演算結果）を用いて、命令 $INST_{rs}$ に応じた演算を行う。これにより、ステップ SH28 では、演算器 120_s の演算結果がステージラッチ回路 110_{s+1} に保持される。このとき、ステージラッチ回路 310_s の値（命令 $INST_{rs}$ ）がステージラッチ回路 310_{s+1} に保持される。以後、第 $s+1$ 演算ステージ～第 $n-1$ 演算ステージ（いずれも図示略）においては、スルー命令 SRC_1 がデコードされることで、演算器 120_s の演算結果がそれぞれの演算ステージにおいて順次スルーされる。また、第 $s+1$ 演算ステージ～第 $n-1$ 演算ステージにおいては、命令 $INST_{rs}$ がステージラッチ回路に順次保持される。

【0251】

そして、第 n 演算ステージにおいて、ステップ SH32 では、命令デコーダ 320_n は、ステージラッチ回路 310_n に命令 $INST_{rs}$ が保持されているため、ステップ SH35 へ進む。なお、ステージラッチ回路 310_n に命令 $INST_n$ が保持されている場合、命令デコーダ 320_n は、ステップ SH33 へ進み、演算器 120_n に対して命令 $INST_n$ をデコードする。これにより、演算器 120_n は、ステージラッチ回路 110_n の値を用いて、命令 $INST_n$ に応じた演算を行う。つぎのステップ SH34 では、演算器 120_n の演算結果がステージラッチ回路 110_{n+1} に保持される。

【0252】

この場合、ステップ SH35 では、命令デコーダ 320_n は、命令 $INST_{rs}$

をスルー命令 SRC_1 に変換した後、ステップ $SH36$ へ進む。ステップ $SH36$ では、命令デコーダ 320_n は、演算器 120_n に対してスルー命令 SRC_1 をデコードする。これにより、演算器 120_n はステージラッチ回路 110_n の値（演算器 120_s 演算結果）をスルーさせる。ステップ $SH37$ では、ステージラッチ回路 110_n の値がステージラッチ回路 110_{n+1} に保持される。つぎのステップ $SH38$ では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_s の演算結果が保持される。

【0253】

つぎに、図 22 および図 23 に示した m 段（第 r 演算ステージ～第 s 演算ステージ）の演算ステージの中に存在する一つの演算ステージ（以下、第 x 演算ステージと称する）における演算器に対して、他の演算器とは独立に演算を実行させるための命令（以下、命令 $INST_x$ と称する）が発行された場合の動作について説明する。以下の説明においては、一例として、上記第 x 演算ステージおよび命令 $INST_x$ を、図 22 に示した第 $r+1$ 演算ステージおよび命令 $INST_{r+1}$ とした場合について説明する。図 24 に示したステップ $SH1$ では、被演算データ用レジスタ 100 、被演算データ用レジスタ 200 および命令用レジスタ 300 に、第 1 被演算データ $SOURCE_1$ 、第 2 被演算データ $SOURCE_2$ および命令 $INST_{r+1}$ がそれぞれ保持される。これにより、上記第 1 被演算データ $SOURCE_1$ 、第 2 被演算データ $SOURCE_2$ および命令 $INST_{r+1}$ がステージラッチ回路 110_1 、ステージラッチ回路 210_1 およびステージラッチ回路 310_1 にそれぞれ保持される。

【0254】

つぎのステップ $SH2$ では、命令デコーダ 320_{11} および命令デコーダ 320_{12} は、ステージラッチ回路 310_1 に命令 $INST_{r+1}$ が保持されているため、ステップ $SH5$ へ進む。ステップ $SH5$ ～ステップ $SH7$ においては、前述と同様にして、マルチプレクサ 700_1 およびマルチプレクサ 800_1 によりステージラッチ回路 110_1 およびステージラッチ回路 210_1 が選択されることで、第 1 被演算データ $SOURCE_1$ および第 2 被演算データ $SOURCE_2$ がステージラッチ回路 110_2 およびステージラッチ回路 210_2 にそれぞれ保持さ

れる。また、ステージラッチ回路 310_2 には、命令 $INST_{r+1}$ が保持される。

【0255】

以後、第2演算ステージ～第 $r-2$ 演算ステージ（いずれも図示略）においては、第1演算ステージの場合と同様にして、マルチプレクサによりステージラッチ回路が選択されることで、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ がそれぞれの演算ステージをスルーで順次通過する。また、第2演算ステージ～第 $r-2$ 演算ステージにおいては、命令 $INST_{r+1}$ がステージラッチ回路に順次保持される。

【0256】

そして、第 $r-1$ 演算ステージにおいて、ステップSH8では、命令デコーダ 320_{r-11} および命令デコーダ 320_{r-12} は、ステージラッチ回路 310_{r-1} に命令 $INST_{r+1}$ が保持されているため、ステップSH11へ進む。ステップSH11～ステップSH13では、前述と同様にして、マルチプレクサ 700_{r-1} およびマルチプレクサ 800_{r-1} によりステージラッチ回路 110_{r-1} およびステージラッチ回路 210_{r-1} が選択されることで、第1被演算データ $SOURCE_1$ および第2被演算データ $SOURCE_2$ が、ステージラッチ回路 110_r およびステージラッチ回路 210_r にそれぞれ保持される。このとき、ステージラッチ回路 310_r には、ステージラッチ回路 110_{r-1} の値（命令 $INST_{r+1}$ ）が保持される。

【0257】

つぎの第 r 演算ステージにおいて、ステップSH14（図25参照）では、命令デコーダ 320_r は、ステージラッチ回路 310_r に命令 $INST_{r+1}$ が保持されているため、ステップSH17へ進む。ステップSH17～ステップSH19においては、前述と同様にして、命令 $INST_{r+1}$ がスルー命令 SRC_1 に変換されることで、ステージラッチ回路 110_r の値（第1被演算データ $SOURCE_1$ ）がステージラッチ回路 110_{r+1} に保持される。このとき、ステージラッチ回路 310_r の値（命令 $INST_{r+1}$ ）がステージラッチ回路 310_{r+1} に保持される。

【0 2 5 8】

つぎの第 $r + 1$ 演算ステージにおいて、ステップ SH 2 0 では、命令デコーダ $3 2 0_{r+1}$ は、ステージラッチ回路 $3 1 0_{r+1}$ に命令 $INST_{r+1}$ が保持されているため、ステップ SH 2 1 へ進む。ステップ SH 2 1 では、命令デコーダ $3 2 0_{r+1}$ は、命令 $INST_{r+1}$ を演算器 $1 2 0_{r+1}$ に対してデコードする。これにより、演算器 $1 2 0_{r+1}$ はステージラッチ回路 $1 1 0_{r+1}$ の値（第 1 被演算データ $SOURCE_1$ ）を用いて、命令 $INST_{r+1}$ に応じた演算を行う。これにより、ステップ SH 2 2 では、演算器 $1 2 0_{r+1}$ の演算結果がステージラッチ回路 $1 1 0_{r+2}$ に保持される。このとき、ステージラッチ回路 $3 1 0_{r+1}$ の値（命令 $INST_{r+1}$ ）がステージラッチ回路 $3 1 0_{r+2}$ に保持される。以後、第 $r + 2$ 演算ステージ～第 $s - 1$ 演算ステージ（いずれも図示略）においては、それぞれの演算ステージにおいてスルー命令 SRC_1 がデコードされることで、演算器 $1 2 0_{r+1}$ の演算結果が順次スルーされる。

【0 2 5 9】

そして、図 2 3 に示した第 s 演算ステージにおいて、ステップ SH 2 6（図 2 6 参照）では、命令デコーダ $3 2 0_s$ は、ステージラッチ回路 $3 1 0_s$ に命令 $INST_{r+1}$ が保持されているため、ステップ SH 2 9 へ進む。ステップ SH 2 9～ステップ SH 3 1 においては、前述と同様にして、スルー命令 SRC_1 がデコードされることで、ステージラッチ回路 $1 1 0_s$ の値（演算器 $1 2 0_{r+1}$ の演算結果）がステージラッチ回路 $1 1 0_{s+1}$ に保持される。以後、第 $s + 1$ 演算ステージ～第 $n - 1$ 演算ステージ（いずれも図示略）においては、スルー命令 SRC_1 がデコードされることで、演算器 $1 2 0_{r+1}$ の演算結果がそれぞれの演算ステージにおいて順次スルーされる。また、第 $s + 1$ 演算ステージ～第 $n - 1$ 演算ステージにおいては、命令 $INST_{r+1}$ がステージラッチ回路に順次保持される。

【0 2 6 0】

そして、第 n 演算ステージにおいて、ステップ SH 3 2 では、命令デコーダ $3 2 0_n$ は、ステージラッチ回路 $3 1 0_n$ に命令 $INST_{r+1}$ が保持されているため、ステップ SH 3 5 へ進む。ステップ SH 3 5～ステップ SH 3 7 においては、前述と同様にして、スルー命令 SRC_1 がデコードされることで、ステージラ

ッチ回路 110_n の値（演算器 120_{r+1} の演算結果）がステージラッチ回路 110_{n+1} に保持される。つぎのステップ SH 38 では、演算結果用レジスタ 500 には、パイプライン演算装置の演算結果として演算器 120_{r+1} の演算結果が保持される。

【0261】

以上説明したように、実施の形態 8 によれば、第 x 演算ステージにおける演算実行時に、第 $x+1$ 演算ステージ～第 n 演算ステージにおいてスルー命令 SRC_1 をそれぞれデコードすることで第 x 演算ステージにおける演算器 120_x の演算結果をスルーさせるようにしたので、第 $x+1$ 演算ステージ～第 n 演算ステージにおけるハードウェア量および消費電力が低減する。また、実施の形態 8 によれば、第 r 演算ステージ～第 s 演算ステージのうちいずれか一つの演算ステージにおける演算器の演算を、他の演算器とは独立させて行うことができる。

【0262】

なお、上述した実施の形態 8 においては、図 22 および図 23 に示した m 段の第 r 演算ステージ～第 s 演算ステージのうちいずれか一つの演算ステージの演算器で演算を単独実行するように構成した例について説明したが、特定の一つの演算ステージの演算器のみで演算を実行するようにしてもよい。また、実施の形態 8 においては、 m 段の第 r 演算ステージ～第 s 演算ステージのうち、第 $x-p$ ($p \leq s-r$) 演算ステージ～第 x 演算ステージのそれぞれの演算器で演算を単独実行するようにしてもよい。

【0263】

以上本発明にかかる実施の形態 1～8 について図面を参照して詳述してきたが、具体的な構成例はこの実施の形態に限られるものではなく、本発明の要旨を逸脱しない範囲の設計変更等があっても本発明に含まれる。たとえば、実施の形態 1～8 においては、演算ステージの段数、ステージラッチ回路、演算器、命令デコーダ、マルチプレクサの個数、組み合わせとして一例を説明した。したがって、実施の形態 1～8 においては、スルー命令 SRC_1 （スルー命令 SRC_2 ）を用いる構成であれば、必要に応じて、上記段数、個数、組み合わせを変えてもよい。

【 0 2 6 4 】

【発明の効果】

以上説明したように、請求項 1 にかかる発明によれば、第 2 演算手段に対して発行された命令をデコードする際に、第 1 演算手段において、上流のラッチ手段に保持される被演算データをスルーさせるようにしたので、第 1 演算手段と第 2 演算手段との間で、ラッチ手段の共有および配線量の低減が可能となることから、ハードウェア量および消費電力が低減するという効果を奏する。

【 0 2 6 5 】

また、請求項 2 にかかる発明によれば、第 1 演算手段に対して発行された命令をデコードする際に、第 2 演算手段において、上流のラッチ手段に保持される第 1 演算手段の演算結果をスルーさせるようにしたので、第 2 演算手段の下流におけるラッチ手段および配線量の低減が可能となることから、ハードウェア量および消費電力が低減するという効果を奏する。

【 0 2 6 6 】

また、請求項 3 にかかる発明によれば、第 x 演算手段に対して発行された命令をデコードする際に、第 1 演算手段～第 $x - 1$ 演算手段において、上流のラッチ手段に保持される被演算データをスルーさせるようにしたので、第 1 演算ステージ～第 $x - 1$ 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減するという効果を奏する。

【 0 2 6 7 】

また、請求項 4 にかかる発明によれば、第 x 演算手段に対して発行された命令をデコードする際に、第 $x + 1$ 演算手段～第 n 演算手段において、上流のラッチ手段に保持される第 x 演算手段の演算結果をスルーさせるようにしたので、第 $x + 1$ 演算ステージ～第 n 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減するという効果を奏する。

【 0 2 6 8 】

また、請求項 5 にかかる発明によれば、第 2 演算手段に対して発行された命令をデコードする際に、第 1 演算手段において、上流のラッチ手段に保持される被

演算データをスルーさせるようにしたので、第 1 演算手段と第 2 演算手段との間で、ラッチ手段の共有および配線量の低減が可能となることから、ハードウェア量および消費電力が低減するという効果を奏する。さらに、請求項 5 にかかる発明によれば、第 2 演算手段において、第 1 演算手段とは独立して演算を行うことができるという効果を奏する。

【0269】

また、請求項 6 にかかる発明によれば、第 1 演算手段に対して発行された命令をデコードする際に、第 2 演算手段において、上流のラッチ手段に保持される第 1 演算手段の演算結果をスルーさせるようにしたので、第 2 演算手段の下流におけるラッチ手段および配線量の低減が可能となることから、ハードウェア量および消費電力が低減するという効果を奏する。さらに、請求項 6 にかかる発明によれば、第 1 演算手段において、第 2 演算手段とは独立して演算を行うことができるという効果を奏する。

【0270】

また、請求項 7 にかかる発明によれば、第 x 演算手段に対して発行された命令をデコードする際に、第 1 演算手段～第 $x-1$ 演算手段において、上流のラッチ手段に保持される被演算データをスルーさせるようにしたので、第 1 演算ステージ～第 $x-1$ 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減するという効果を奏する。さらに、請求項 7 にかかる発明によれば、第 r 演算手段～第 s 演算手段における第 x 演算手段の演算を、他の演算手段とは独立させて行うことができるという効果を奏する。

【0271】

また、請求項 8 にかかる発明によれば、第 x 演算手段～第 $x+p$ 演算手段に対して発行された命令をデコードする際に、第 1 演算手段～第 $x-1$ 演算手段において、上流のラッチ手段に保持される被演算データをスルーさせるようにしたので、第 1 演算ステージ～第 $x-1$ 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減するという効果を奏する。さらに、請求項 8 にかかる発明によれば、第 r 演算手段～第

s 演算手段における第 x 演算手段～第 x + p 演算手段の演算を、他の演算手段とは独立させて行うことができるという効果を奏する。

【0 2 7 2】

また、請求項 9 にかかる発明によれば、第 x 演算手段に対して発行された命令をデコードする際に、第 x + 1 演算手段～第 n 演算手段において、上流のラッチ手段に保持される第 x 演算手段の演算結果をスルーさせるようにしたので、第 x + 1 演算ステージ～第 n 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減するという効果を奏する。さらに、請求項 9 にかかる発明によれば、第 r 演算手段～第 s 演算手段における第 x 演算手段の演算を、他の演算手段とは独立させて行うことができるという効果を奏する。

【0 2 7 3】

また、請求項 1 0 にかかる発明によれば、第 x - p 演算手段～第 x 演算手段に対して発行された命令をデコードする際に、第 x + 1 演算手段～第 n 演算手段において、上流のラッチ手段に保持される第 x - p 演算手段～第 x 演算手段の演算結果をスルーさせるようにしたので、第 x + 1 演算ステージ～第 n 演算ステージにおけるステージラッチ回路、配線が従来の場合に比して減るため、ハードウェア量および消費電力が低減するという効果を奏する。さらに、請求項 1 0 にかかる発明によれば、第 r 演算手段～第 s 演算手段における第 x - p 演算手段～第 x 演算手段の演算を、他の演算手段とは独立させて行うことができるという効果を奏する。

【図面の簡単な説明】

【図 1】

本発明にかかる実施の形態 1 の構成を示すブロック図である。

【図 2】

同実施の形態 1 の動作を説明するフローチャートである。

【図 3】

本発明にかかる実施の形態 2 の構成を示すブロック図である。

【図 4】

同実施の形態 2 の動作を説明するフローチャートである。

【図 5】

本発明にかかる実施の形態 3 の構成を示すブロック図である。

【図 6】

同実施の形態 3 の動作を説明するフローチャートである。

【図 7】

同実施の形態 3 の動作を説明するフローチャートである。

【図 8】

同実施の形態 3 の動作を説明するフローチャートである。

【図 9】

本発明にかかる実施の形態 4 の構成を示すブロック図である。

【図 1 0】

同実施の形態 4 の動作を説明するフローチャートである。

【図 1 1】

同実施の形態 4 の動作を説明するフローチャートである。

【図 1 2】

同実施の形態 4 の動作を説明するフローチャートである。

【図 1 3】

本発明にかかる実施の形態 5 の構成を示すブロック図である。

【図 1 4】

同実施の形態 5 の動作を説明するフローチャートである。

【図 1 5】

本発明にかかる実施の形態 6 の構成を示すブロック図である。

【図 1 6】

同実施の形態 6 の動作を説明するフローチャートである。

【図 1 7】

本発明にかかる実施の形態 7 の構成を示すブロック図である。

【図 1 8】

同実施の形態 7 の構成を示すブロック図である。

【図 1 9】

同実施の形態 7 の動作を説明するフローチャートである。

【図 2 0】

同実施の形態 7 の動作を説明するフローチャートである。

【図 2 1】

同実施の形態 7 の動作を説明するフローチャートである。

【図 2 2】

本発明にかかる実施の形態 8 の構成を示すブロック図である。

【図 2 3】

同実施の形態 8 の構成を示すブロック図である。

【図 2 4】

同実施の形態 8 の動作を説明するフローチャートである。

【図 2 5】

同実施の形態 8 の動作を説明するフローチャートである。

【図 2 6】

同実施の形態 8 の動作を説明するフローチャートである。

【図 2 7】

従来のパイプライン演算装置の構成を示す図である。

【符号の説明】

- 1 1 0₁、1 1 0₂、1 1 0₃ ステージラッチ回路
- 1 1 0_{x-1}、1 1 0_x、1 1 0_{x+1}、1 1 0_{x+2} ステージラッチ回路
- 1 1 0_{r-1}、1 1 0_r、1 1 0_{r+1}、1 1 0_{r+2} ステージラッチ回路
- 1 1 0_s、1 1 0_{s+1} ステージラッチ回路
- 1 1 0_n、1 1 0_{n+1} ステージラッチ回路
- 1 2 0₁、1 2 0₂ 演算器
- 1 2 0_{x-1}、1 2 0_x、1 2 0_{x+1}、1 2 0_n 演算器
- 1 2 0_{r-1}、1 2 0_r、1 2 0_{r+1}、演算器
- 1 2 0_s 演算器
- 2 1 0₁、2 1 0₂ ステージラッチ回路

210_{x-1} 、 210_x ステージラッチ回路

210_{r-1} 、 210_r ステージラッチ回路

220_{r-1} 演算器

310_1 、 310_2 ステージラッチ回路

310_{x-1} 、 310_x 、 310_{x+1} 、 310_{x+2} ステージラッチ回路

310_{r-1} 、 310_r 、 310_{r+1} 、 310_{r+2} ステージラッチ回路

310_s 、 310_n ステージラッチ回路

320_1 、 320_2 命令デコーダ

320_{r-11} 、 320_{r-12} 、 320_r 、 320_{r+1} 命令デコーダ

320_s 、 320_n 命令デコーダ

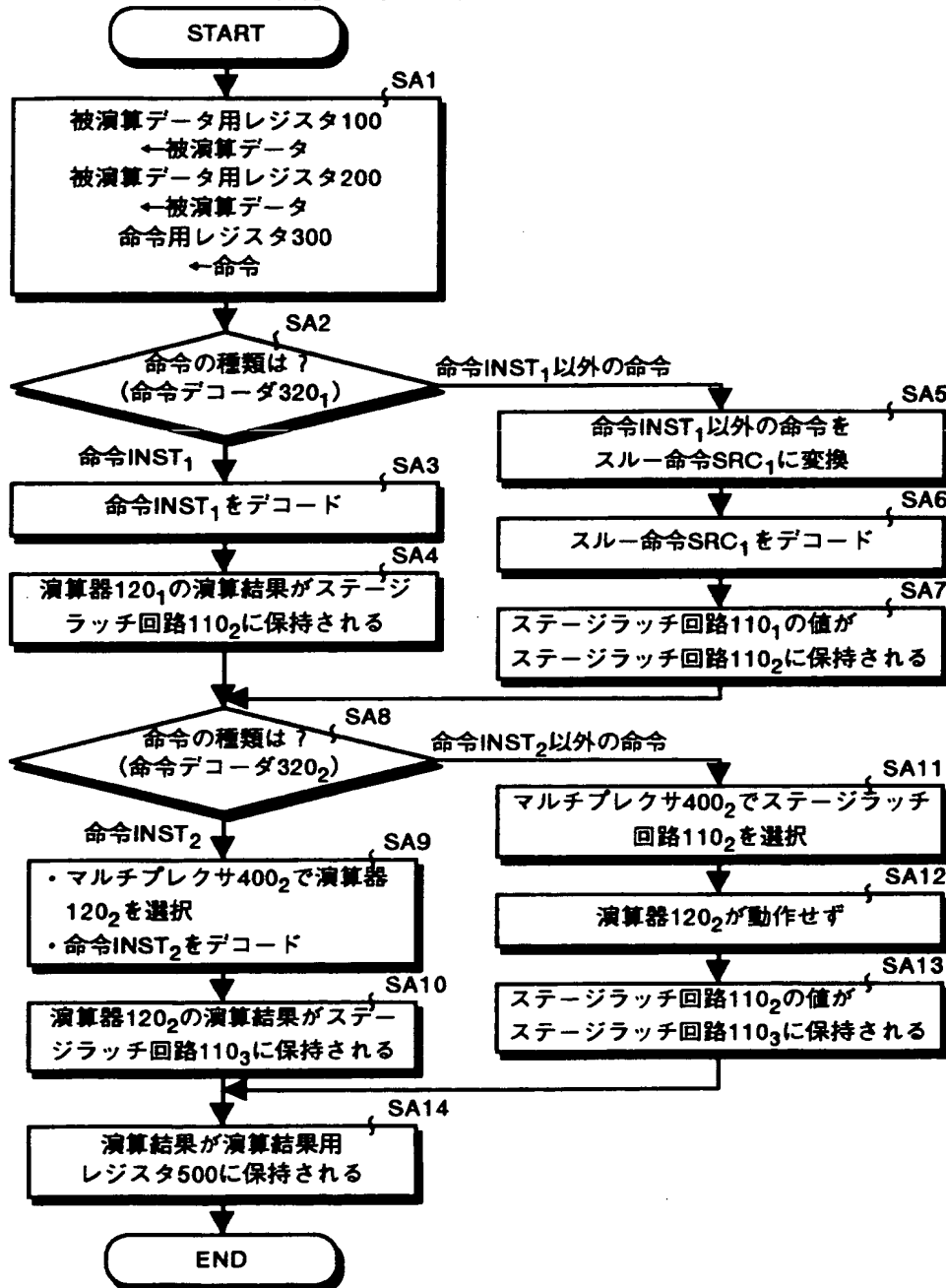
【図 1】

実施の形態 1 の構成を示すブロック図



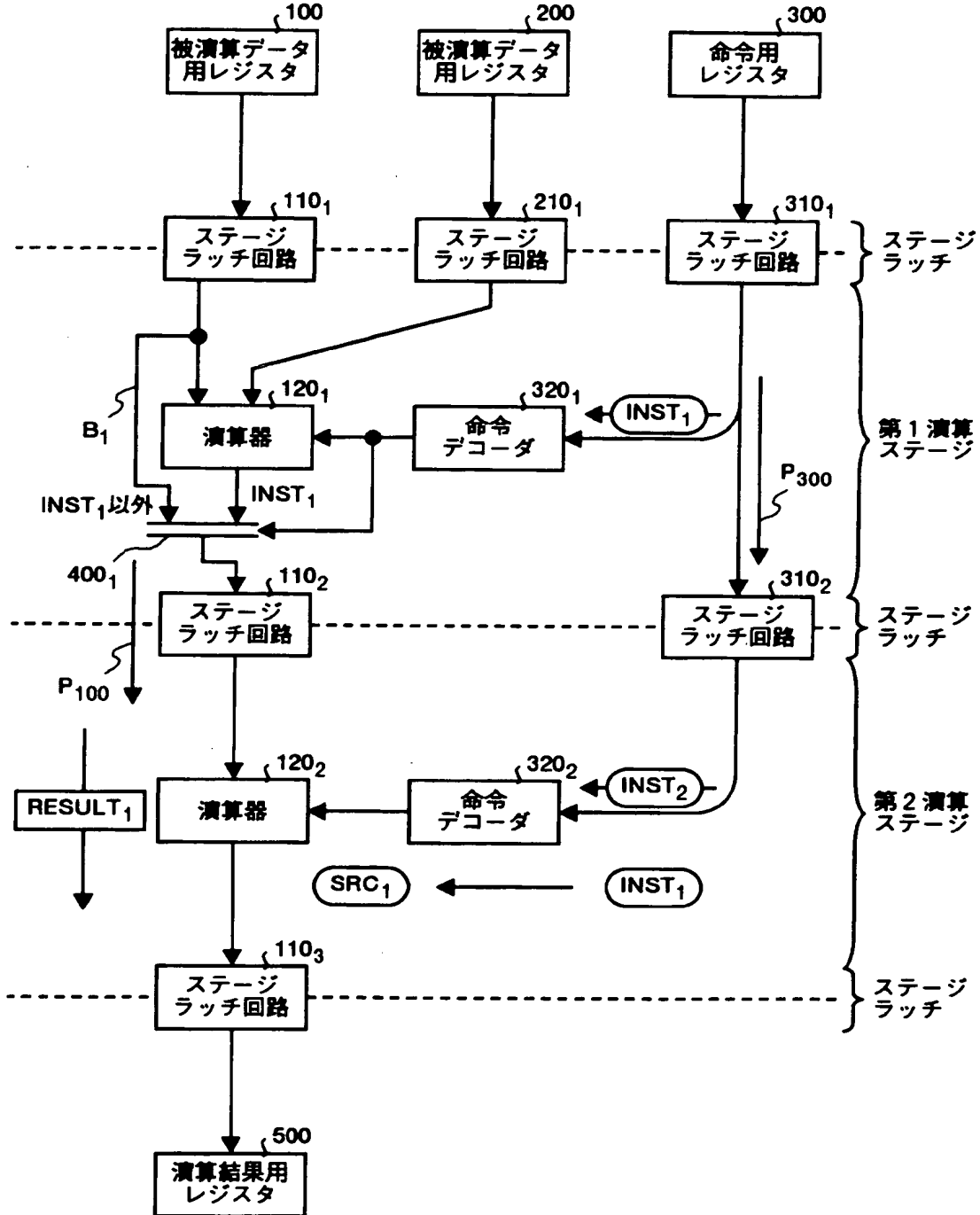
【図 2】

実施の形態 1 の動作を説明するフローチャート



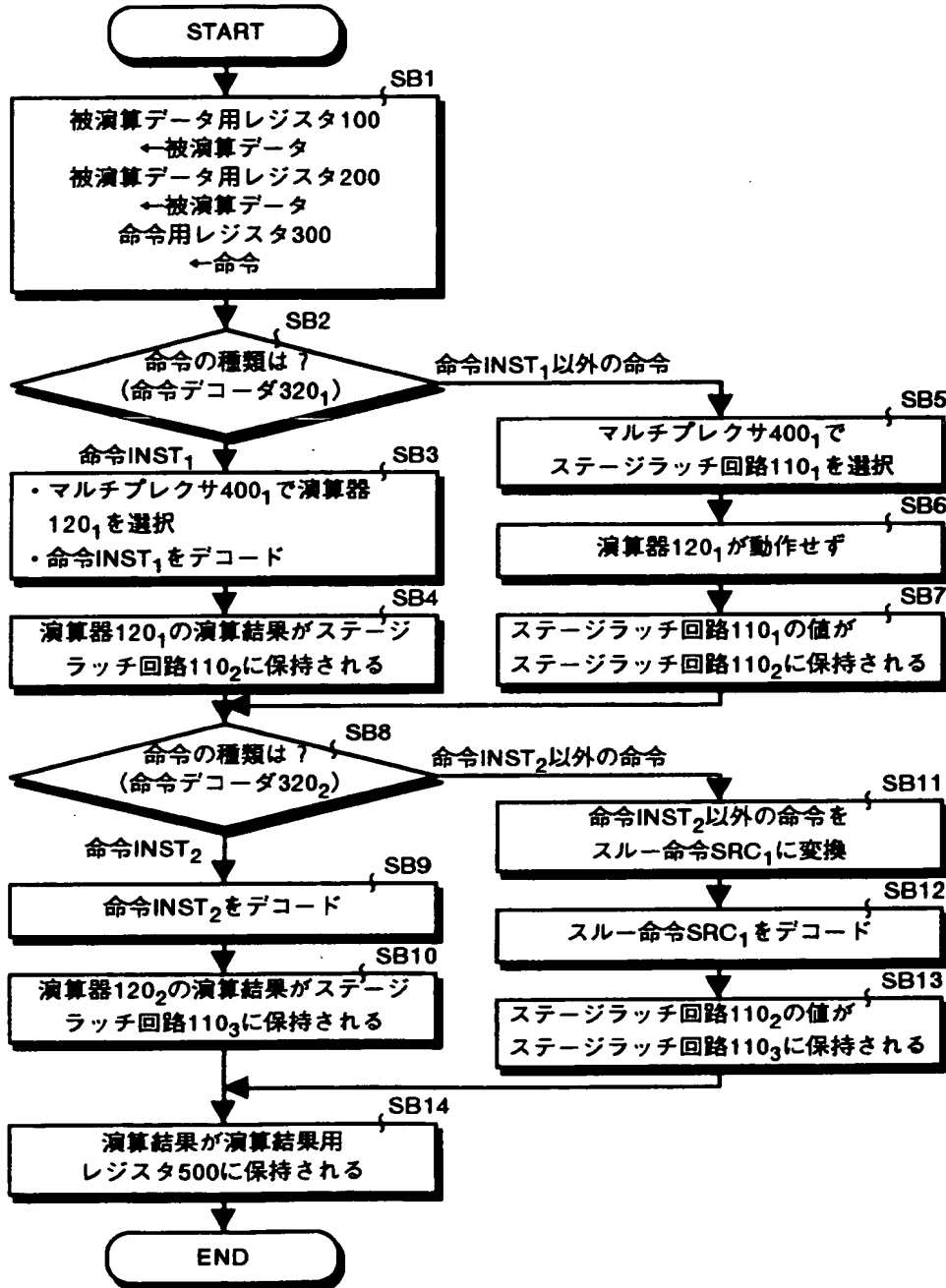
【図 3】

実施の形態 2 の構成を示すブロック図



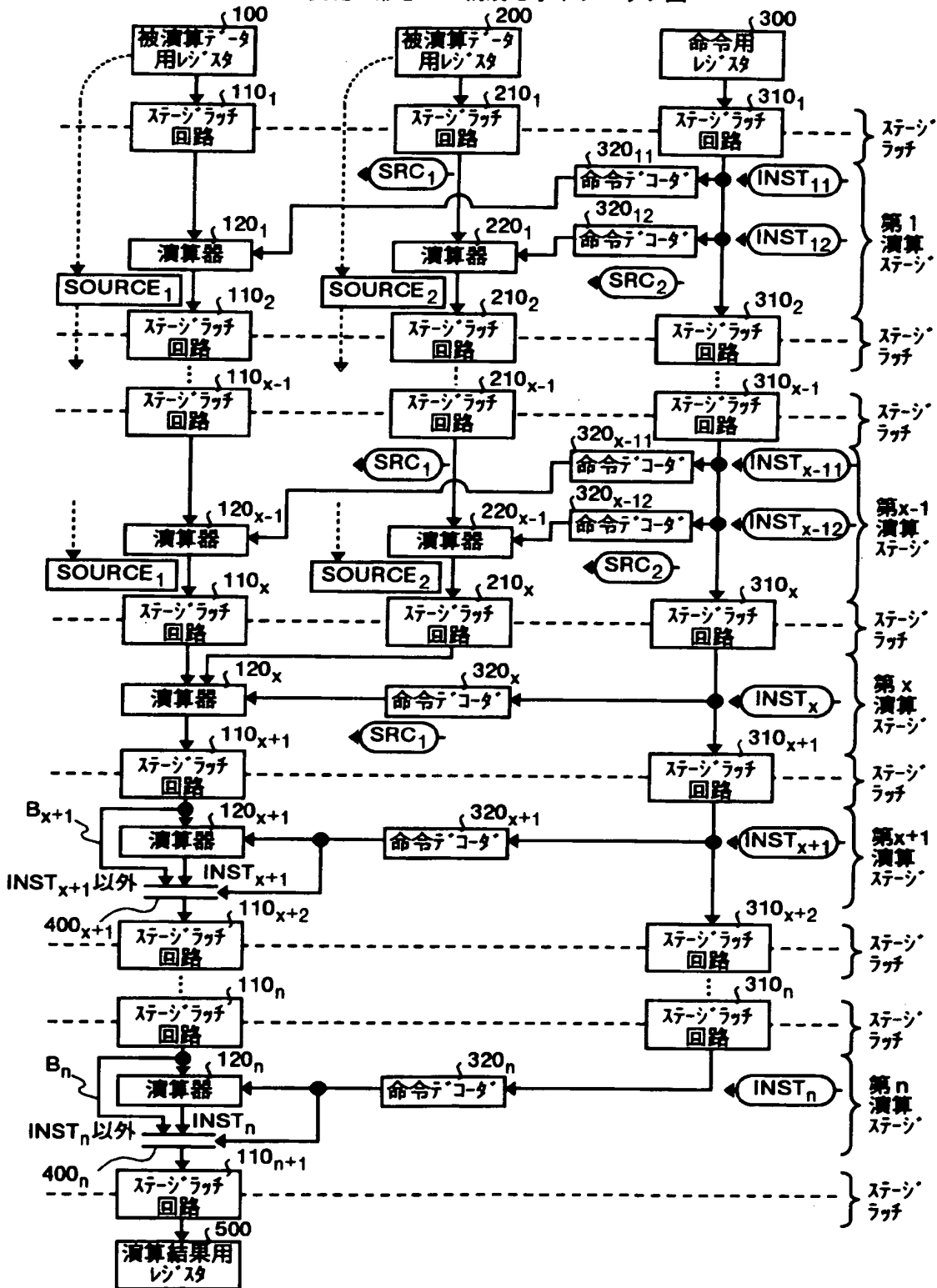
【図 4】

実施の形態 2 の動作を説明するフローチャート



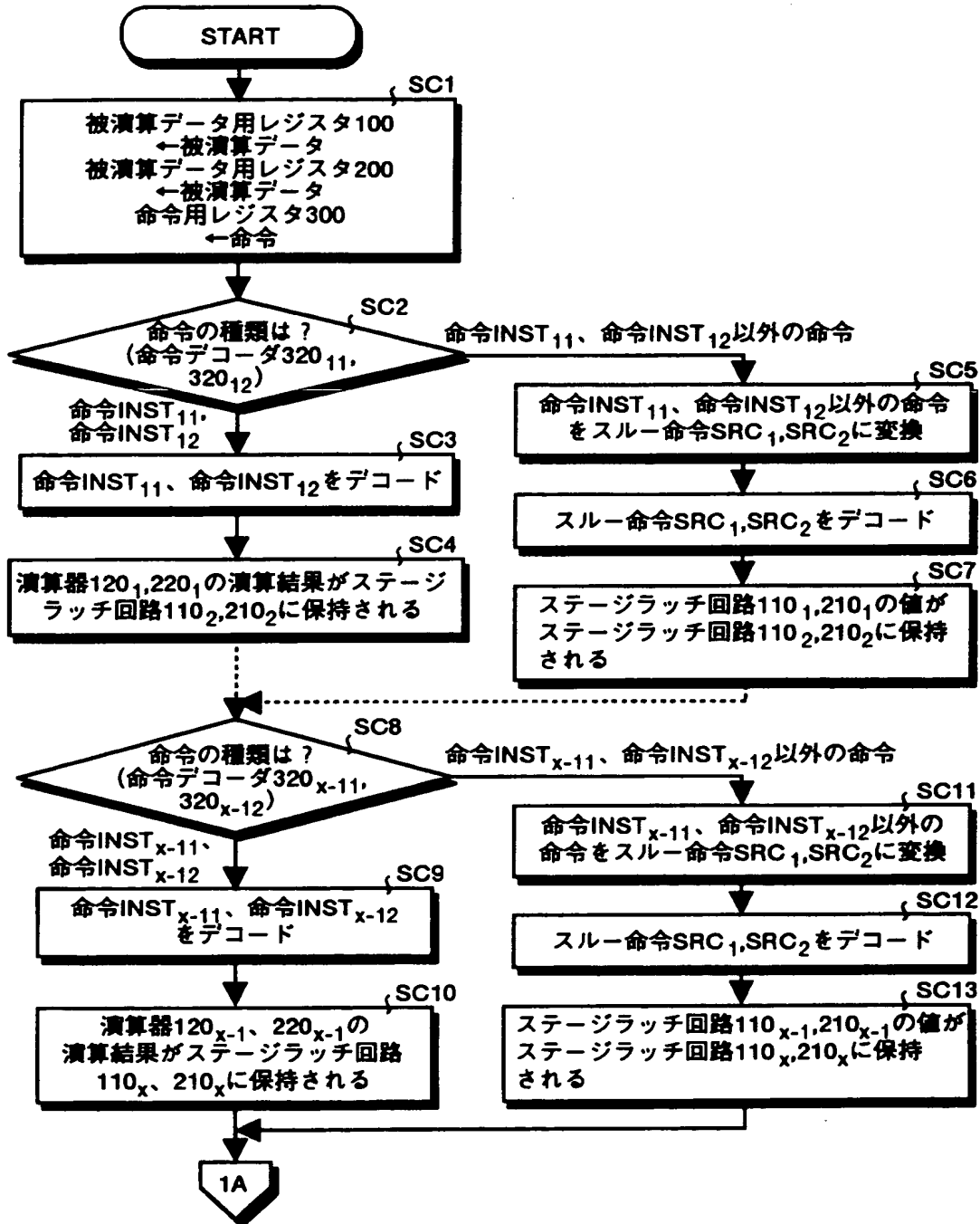
【図 5】

実施の形態 3 の構成を示すブロック図



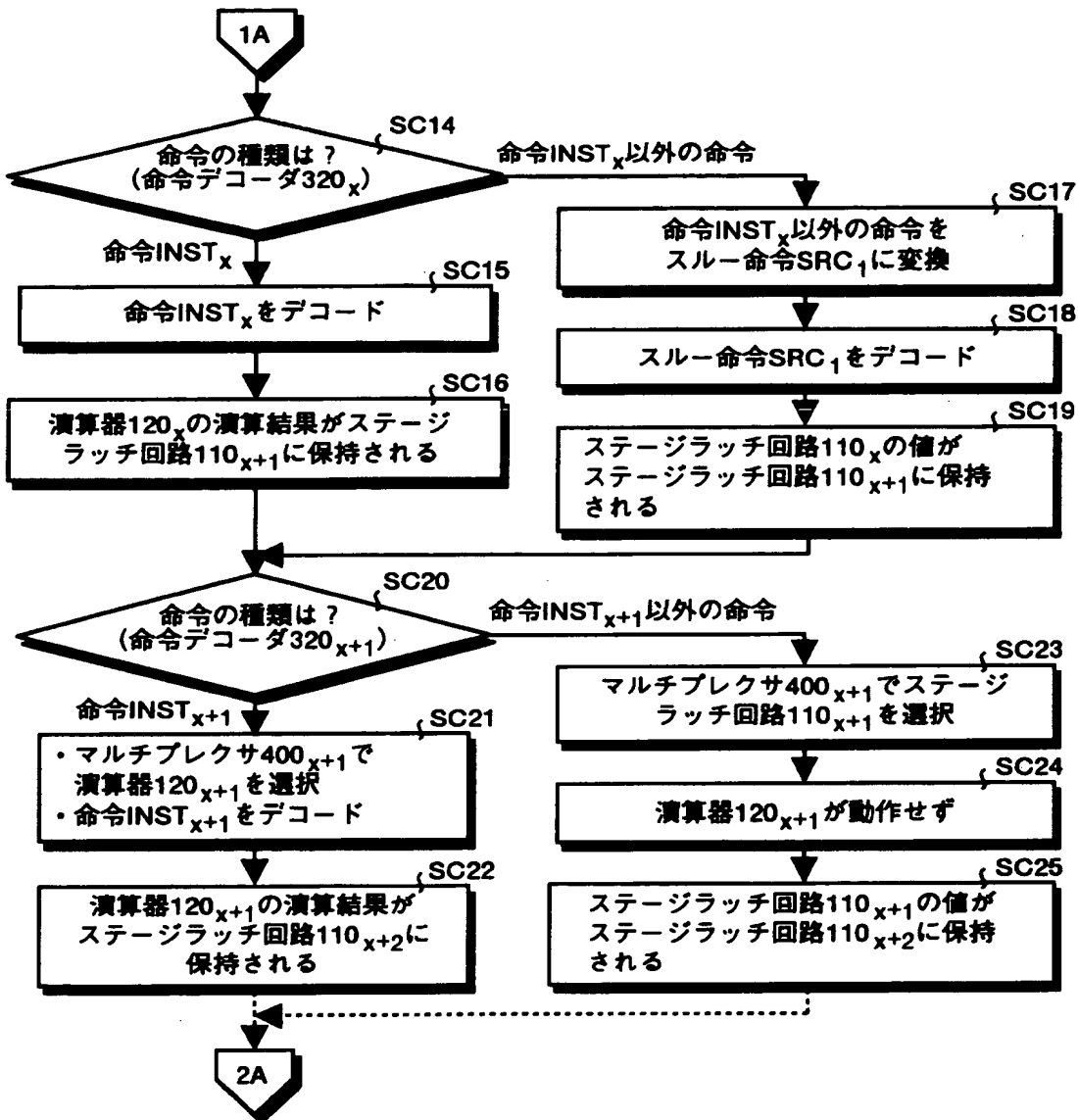
【図 6】

実施の形態 3 の動作を説明するフローチャート



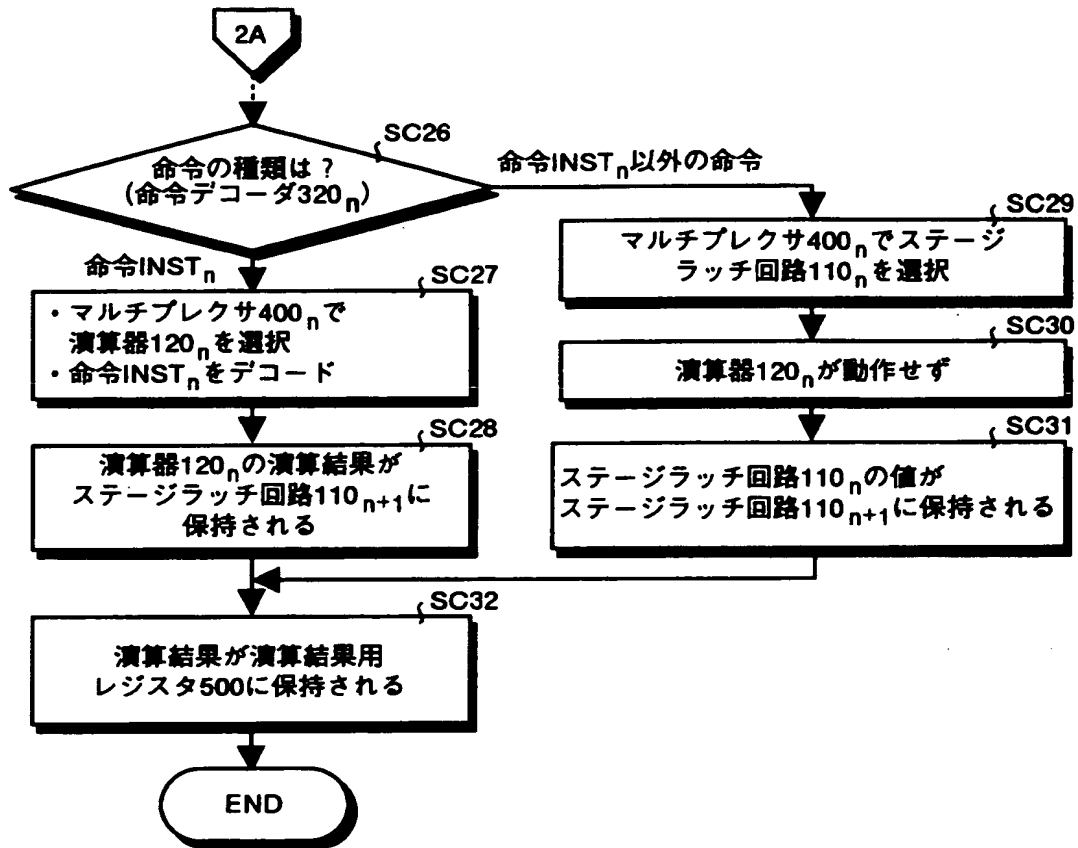
【図 7】

実施の形態 3 の動作を説明するフローチャート



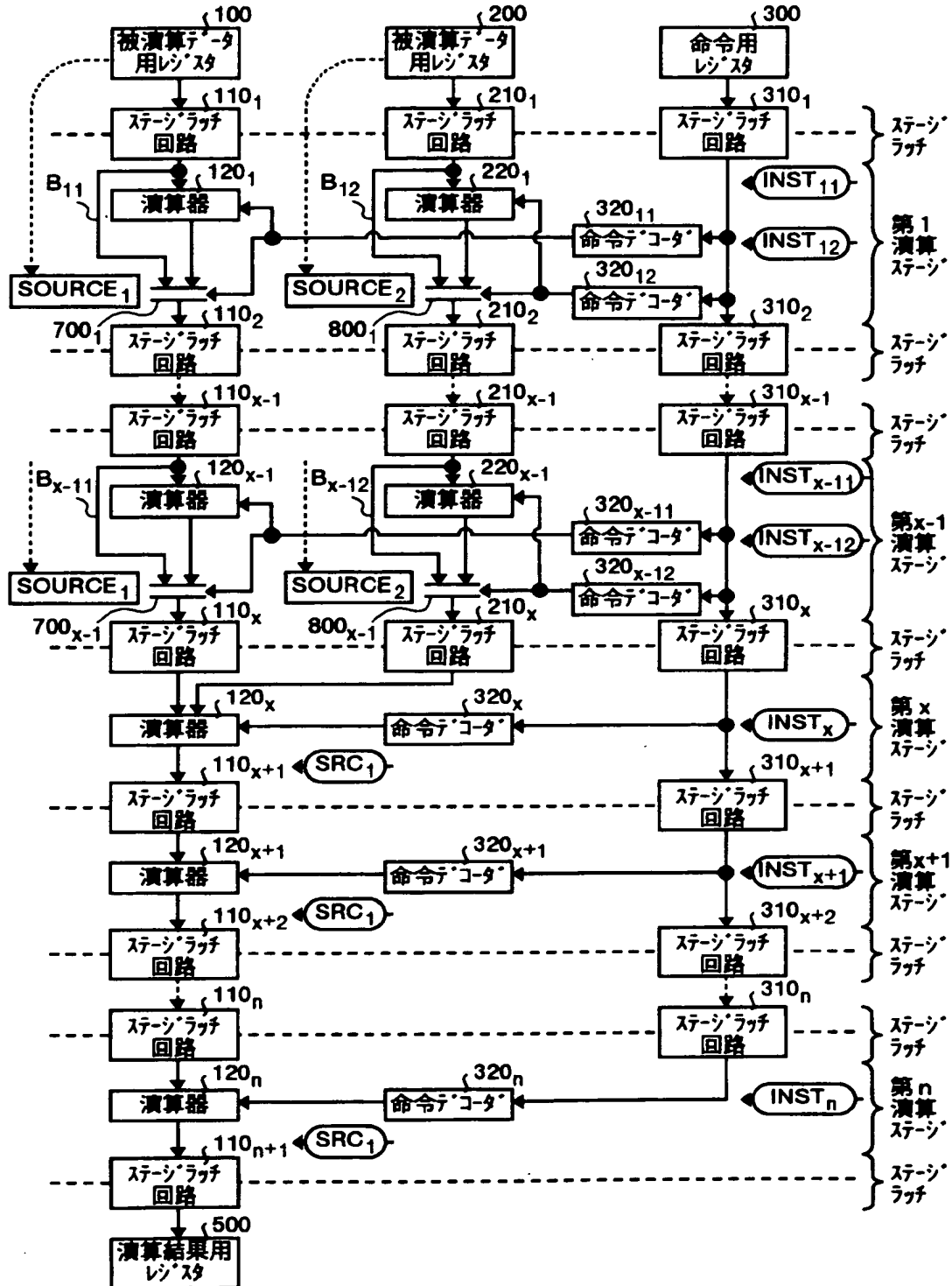
【図 8】

実施の形態 3 の動作を説明するフローチャート

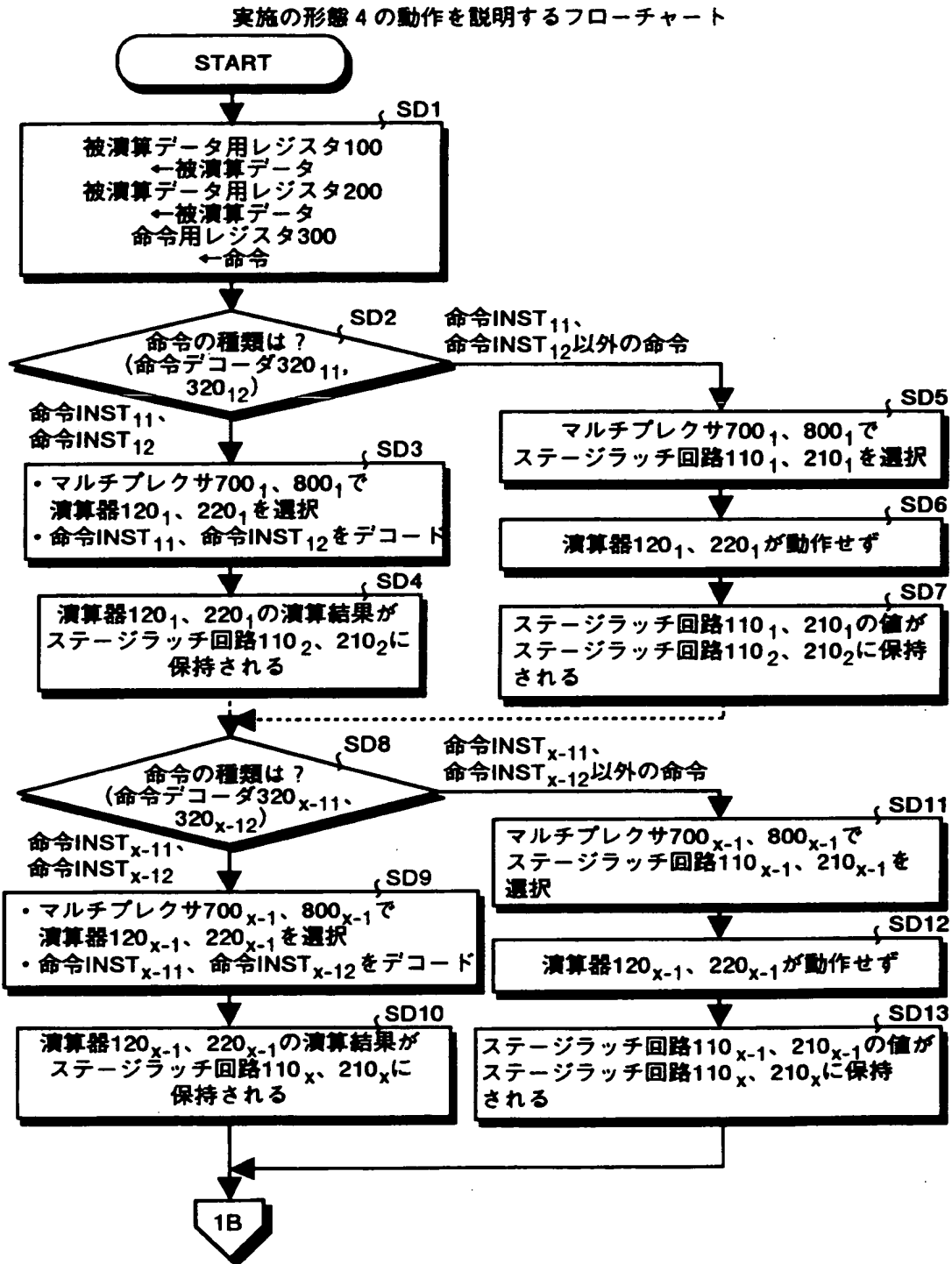


【図 9】

実施の形態 4 の構成を示すブロック図

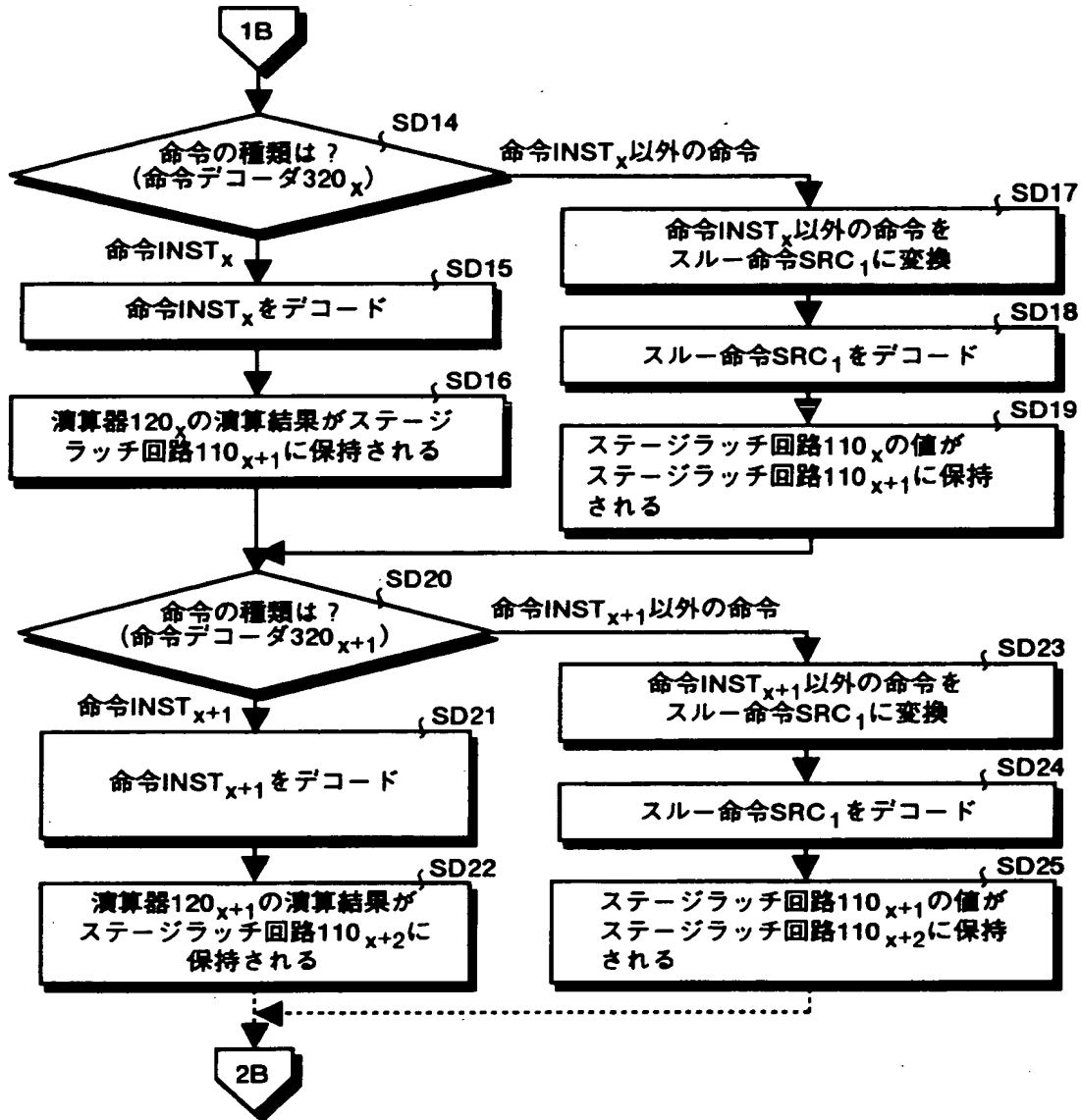


【図 1 0】



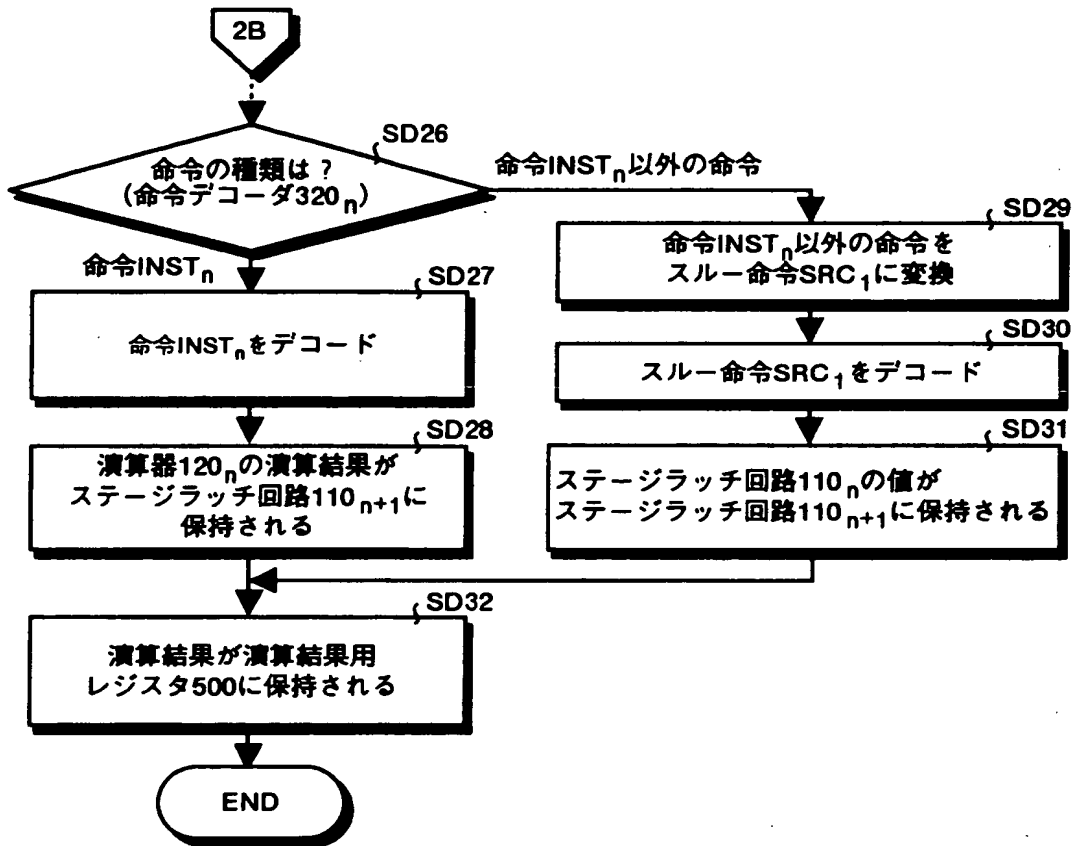
【図 1 1】

実施の形態 4 の動作を説明するフローチャート



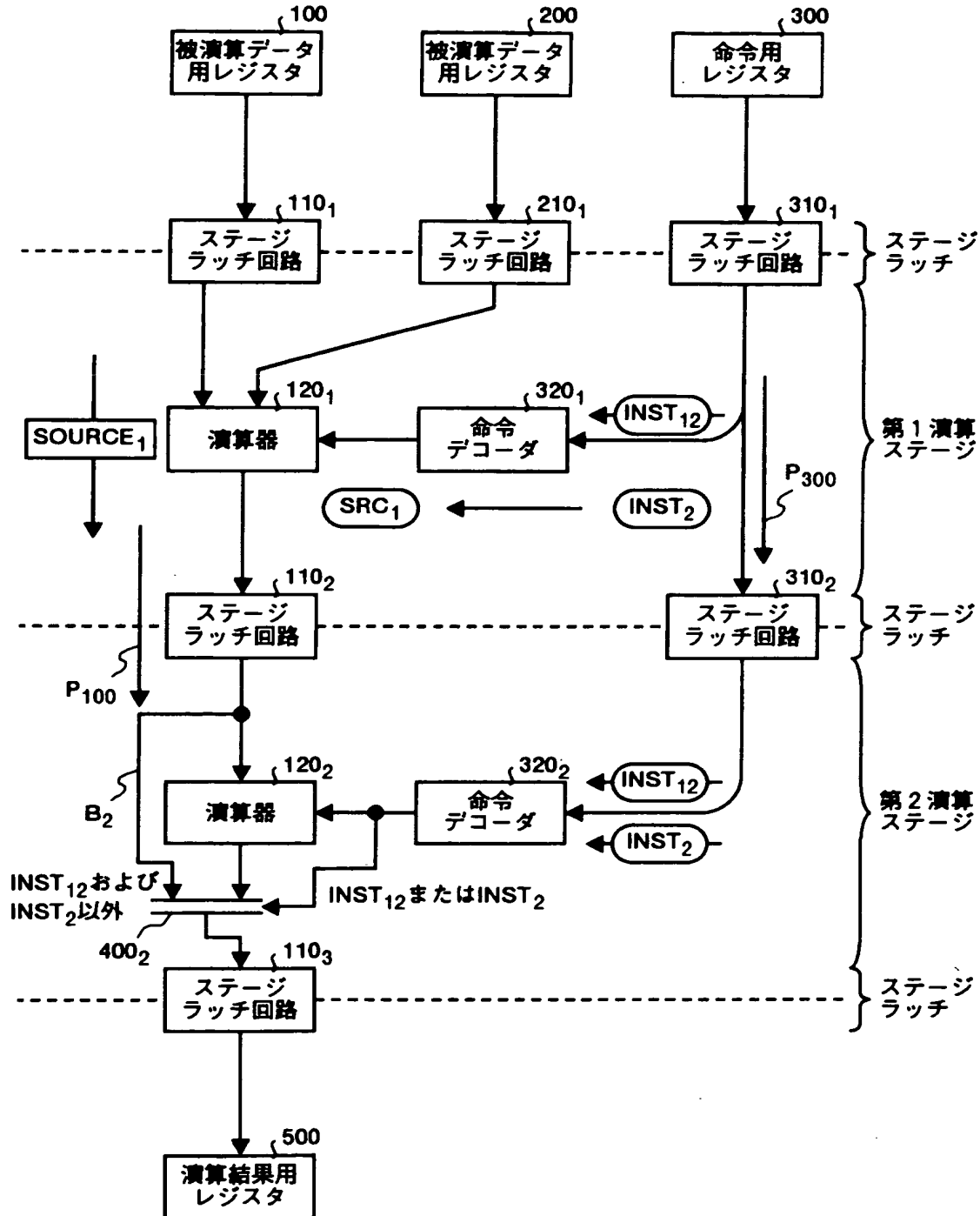
【図 1 2】

実施の形態 4 の動作を説明するフローチャート

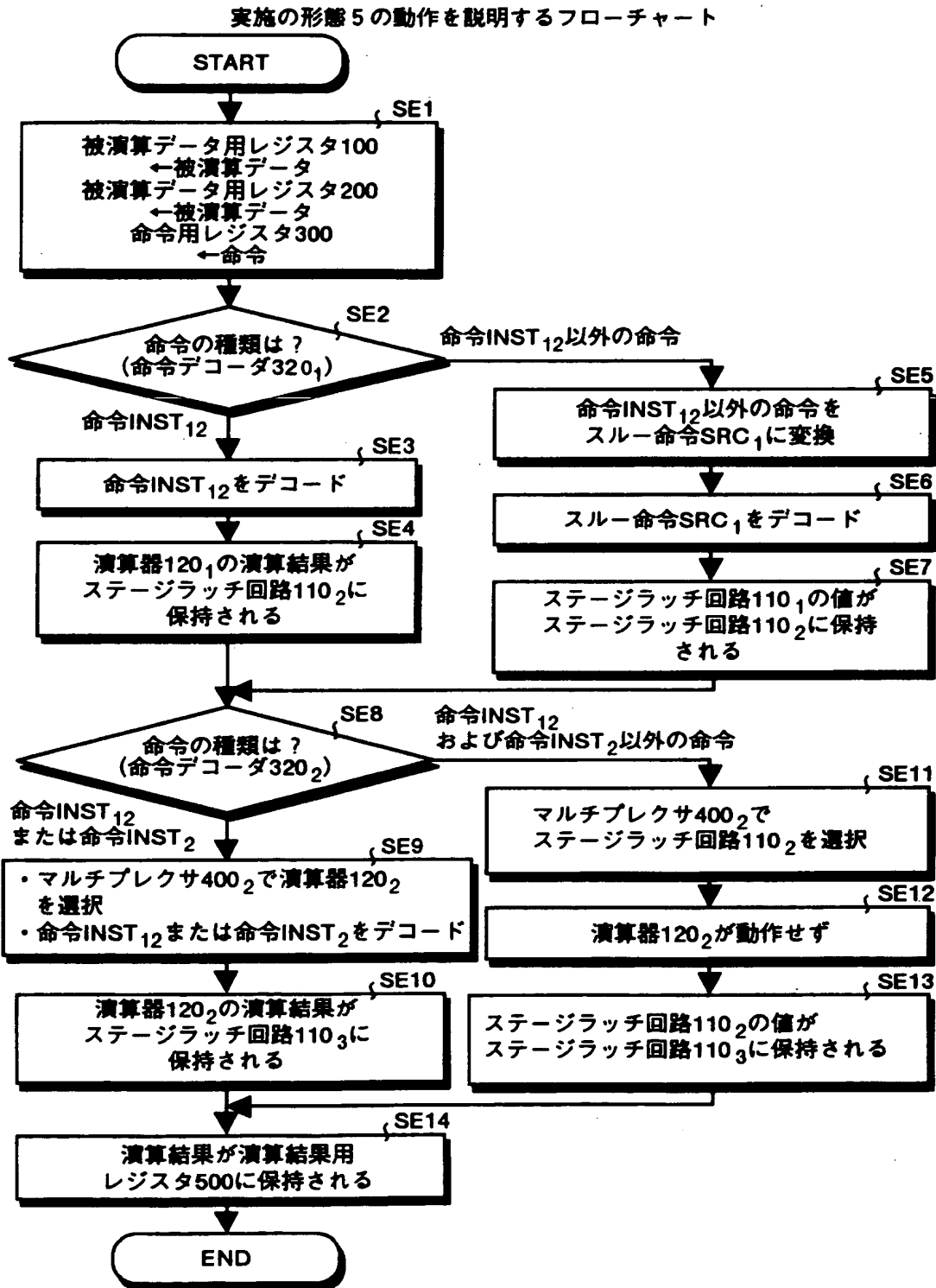


【図 1 3】

実施の形態 5 の構成を示すブロック図

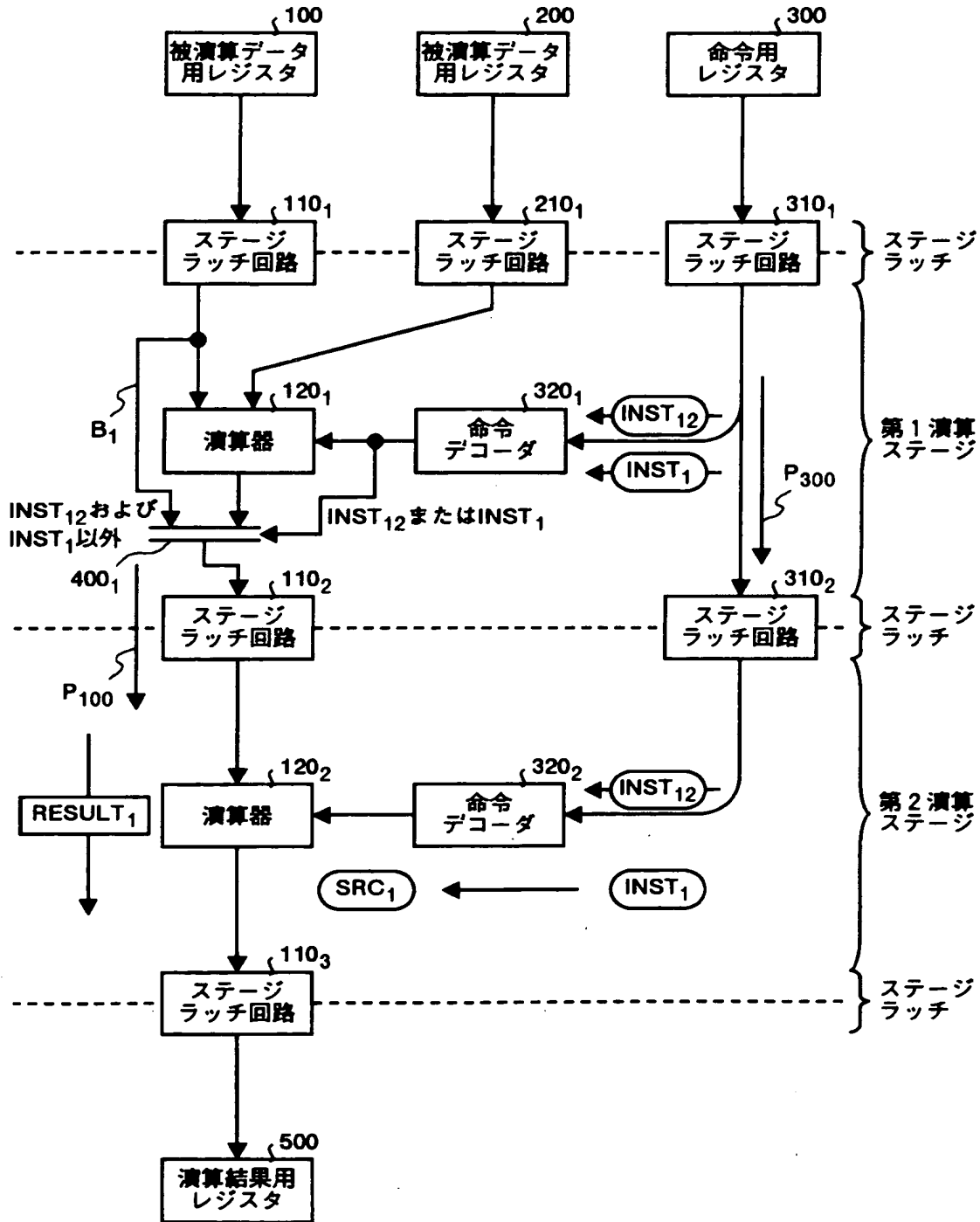


【図 1 4】

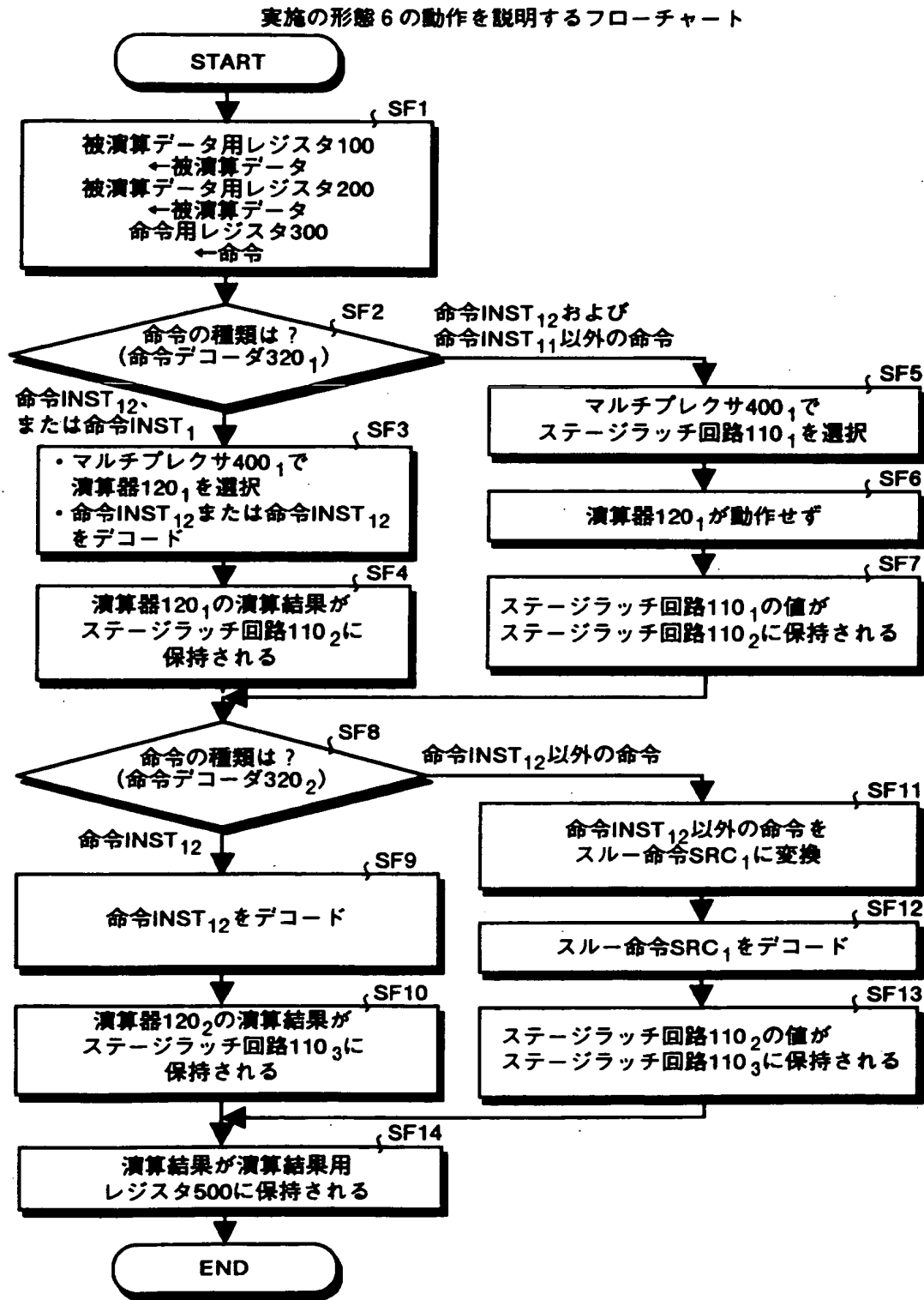


【図 1 5】

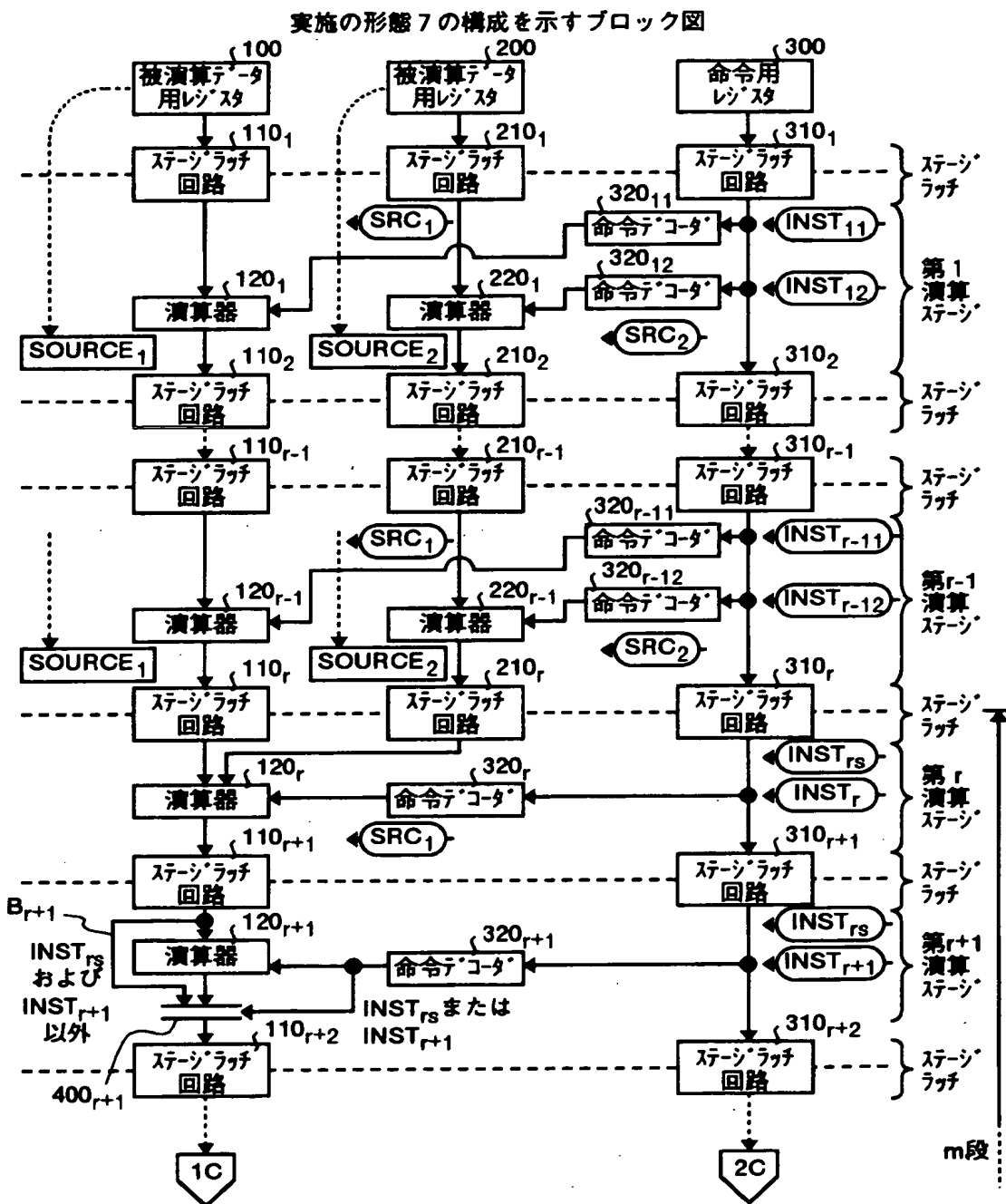
実施の形態 6 の構成を示すブロック図



【図 1 6】

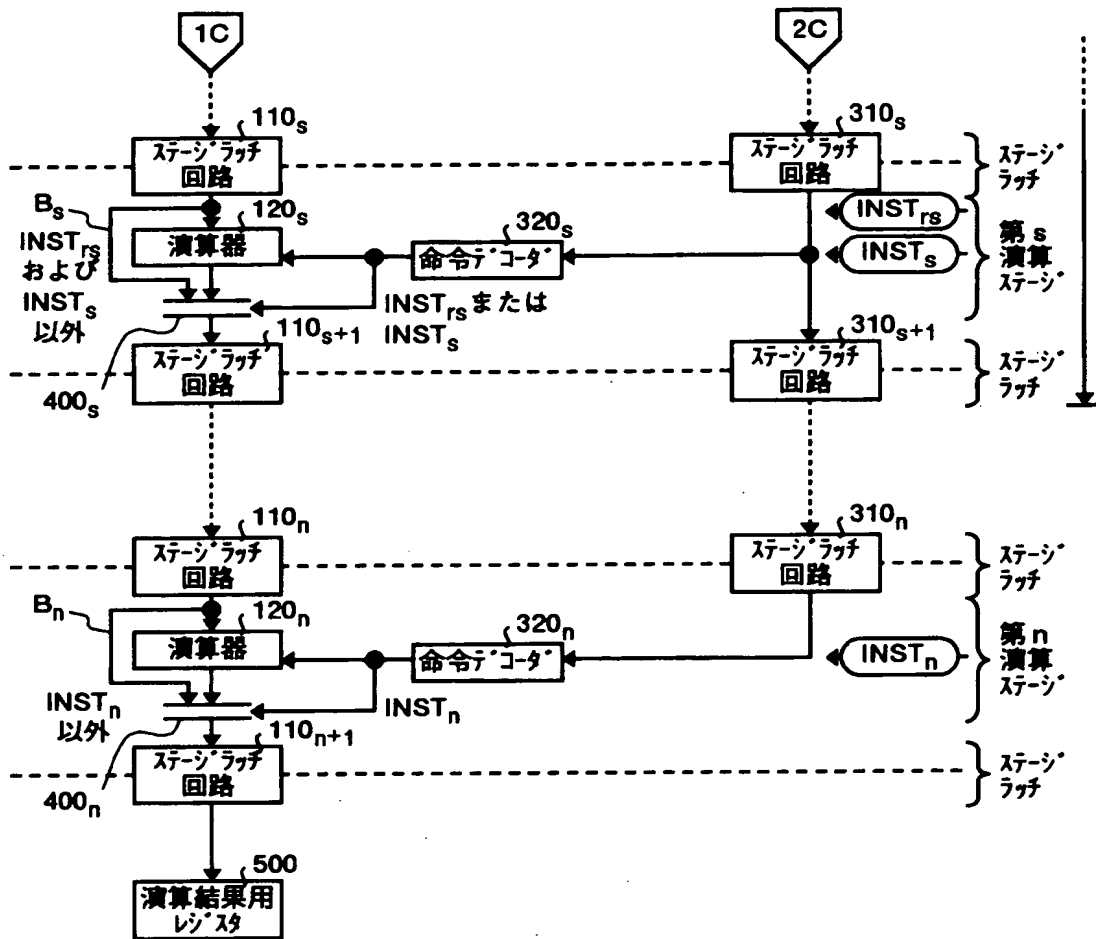


【図 1 7】

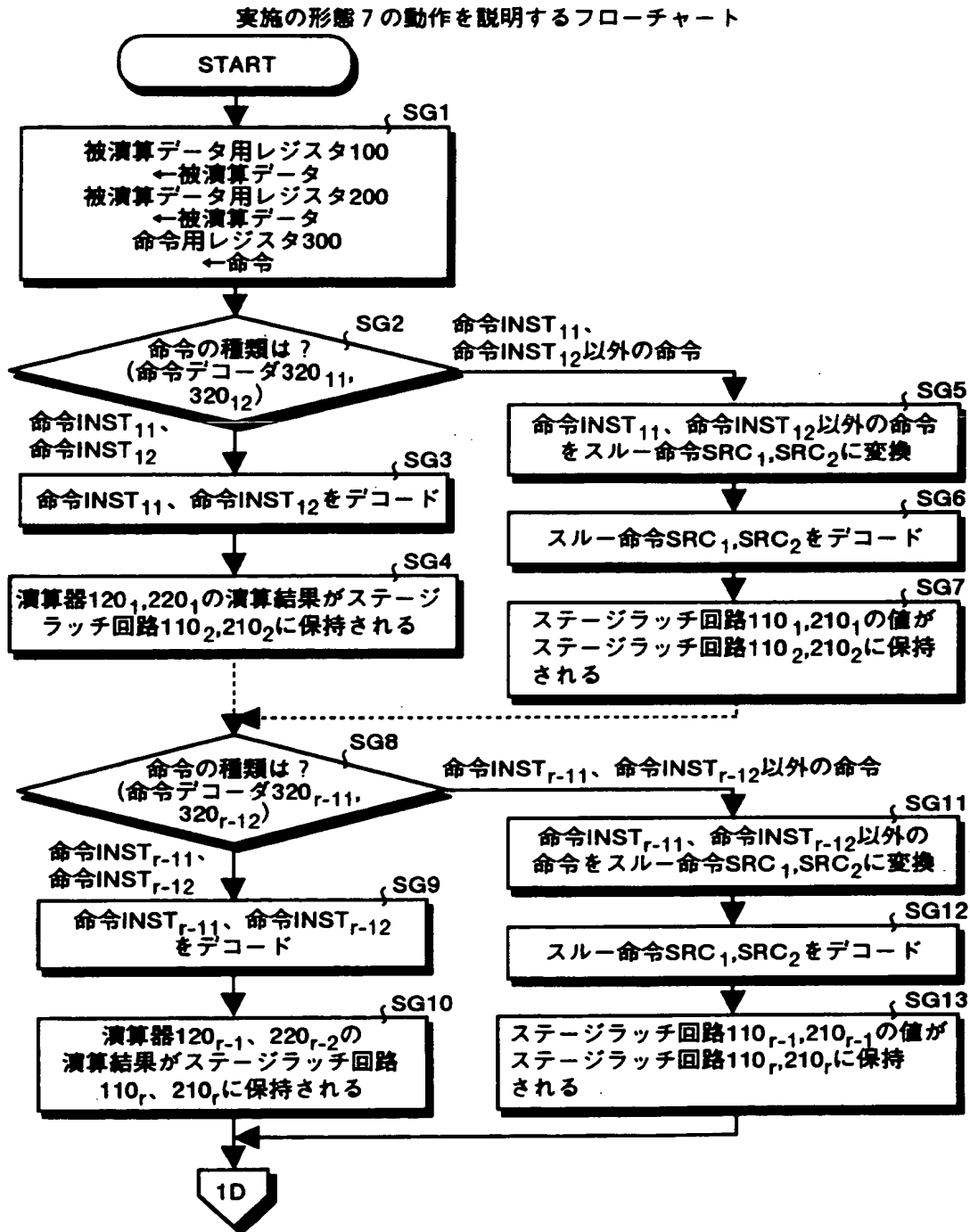


【図 1 8】

実施の形態 7 の構成を示すブロック図

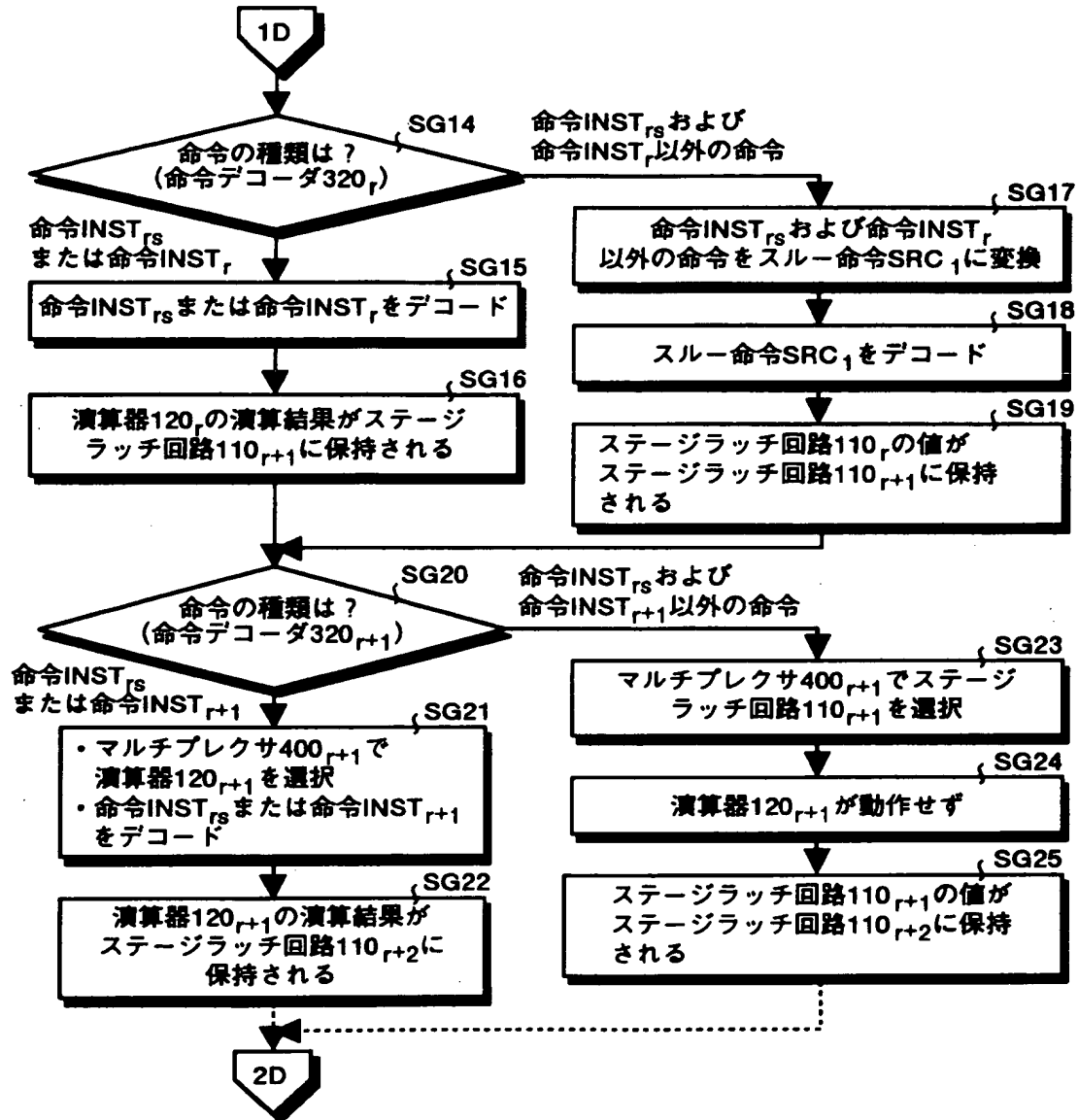


【図 1 9】



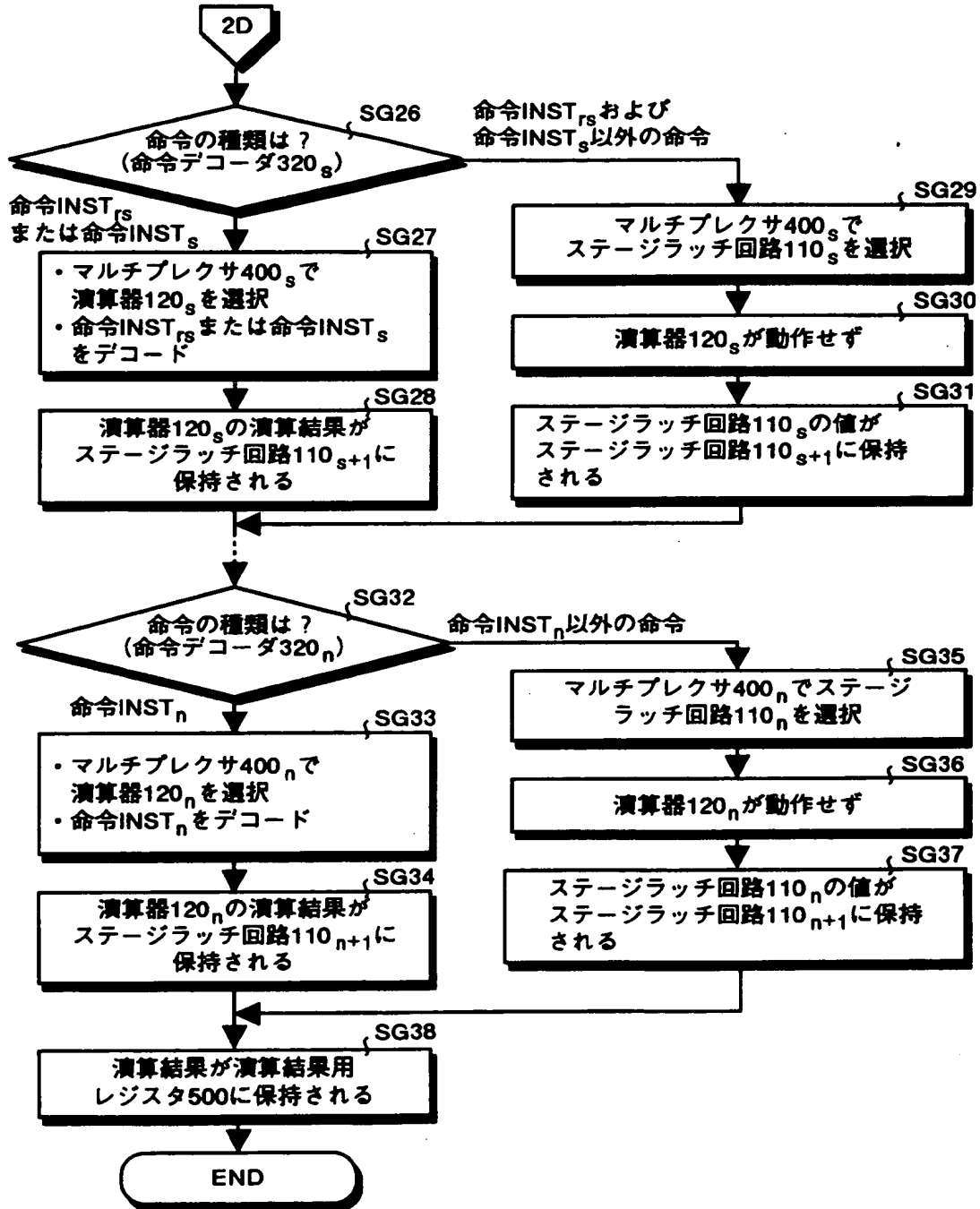
【図 20】

実施の形態 7 の動作を説明するフローチャート

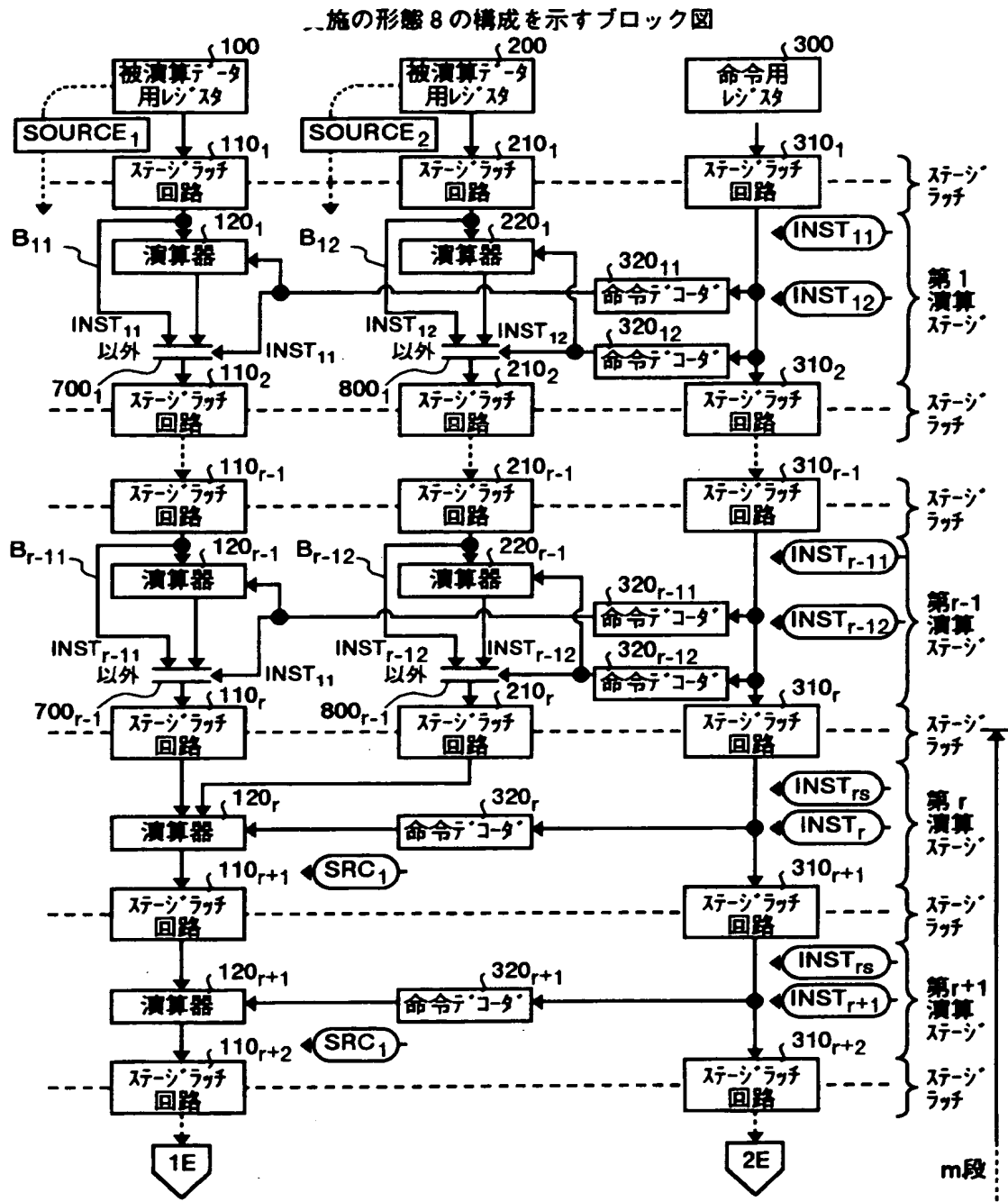


【図 2 1】

実施の形態 7 の動作を説明するフローチャート

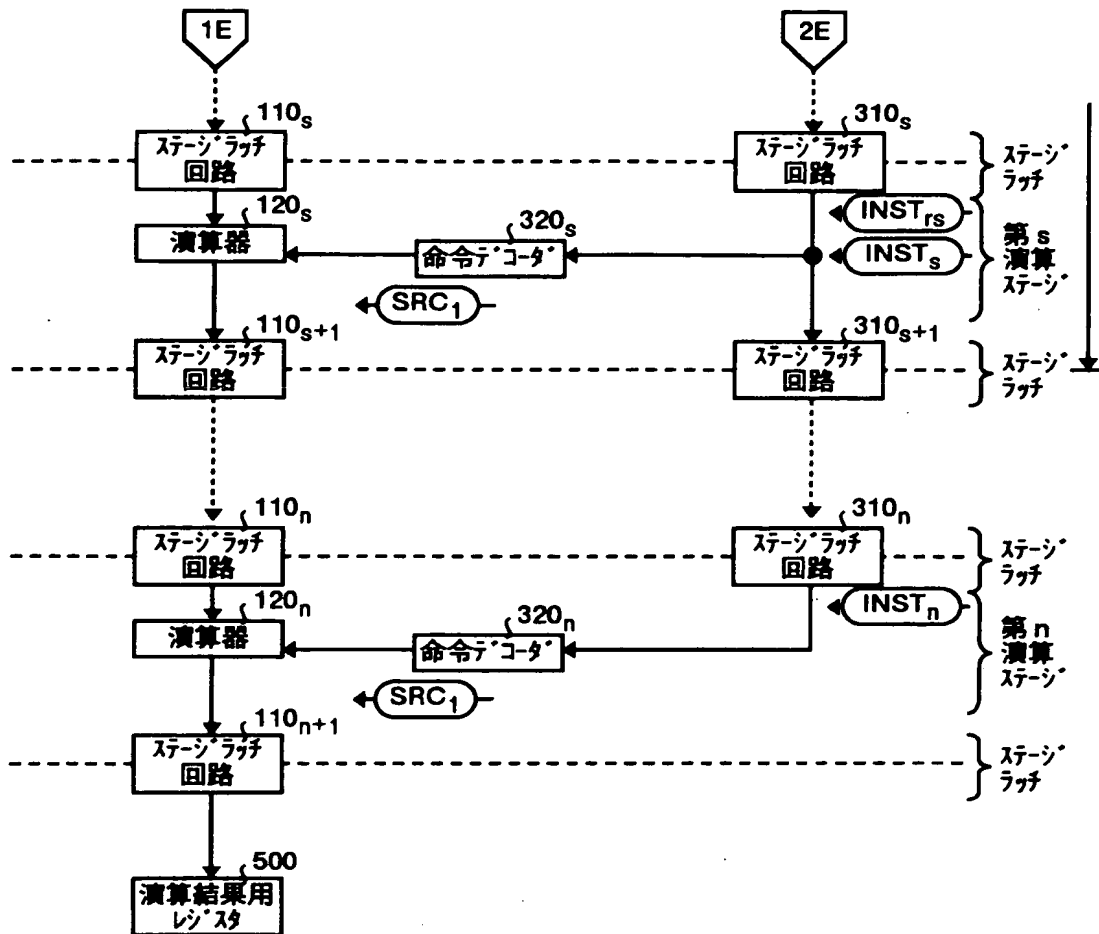


【図 2 2】

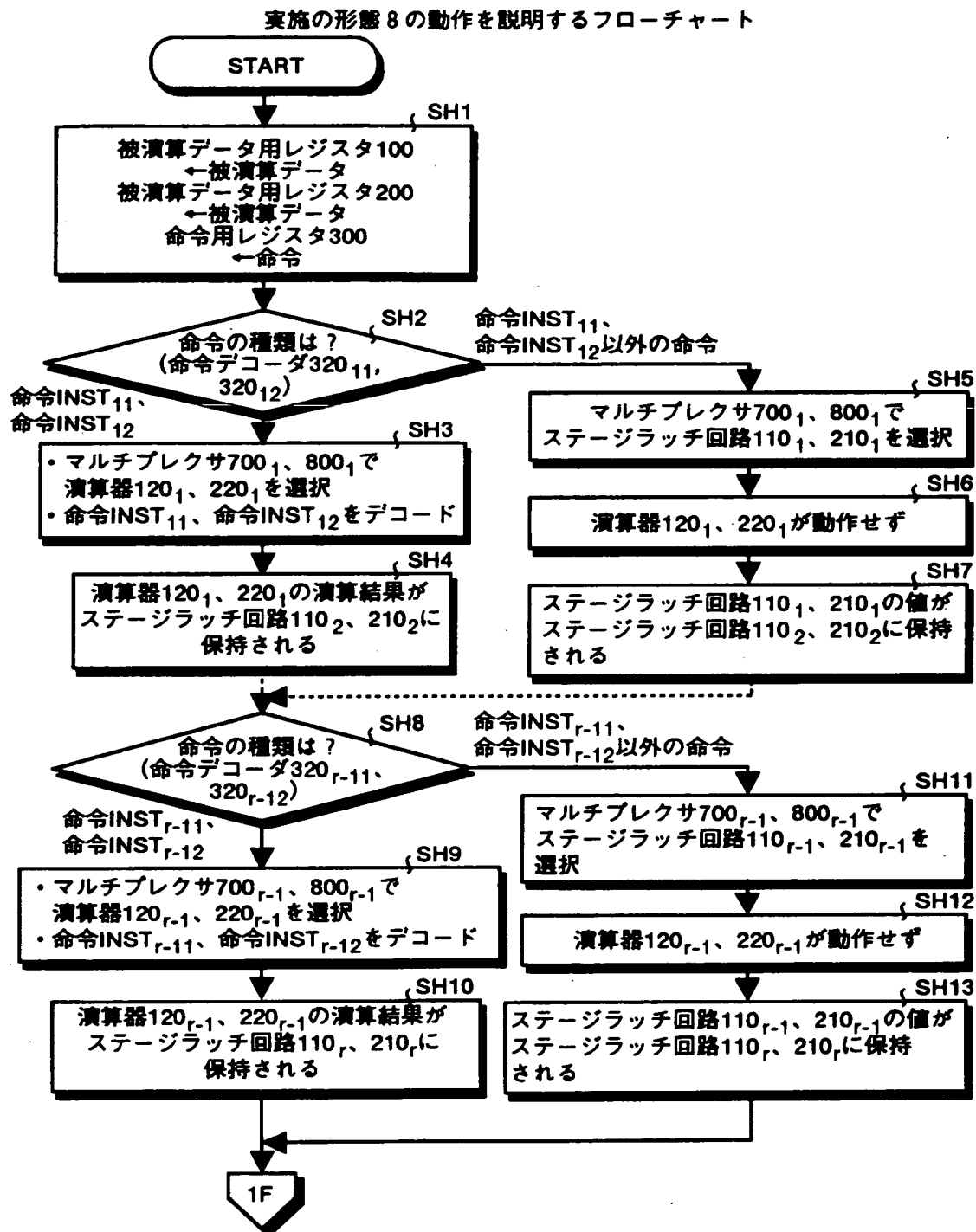


【図 2 3】

実施の形態 8 の構成を示すブロック図

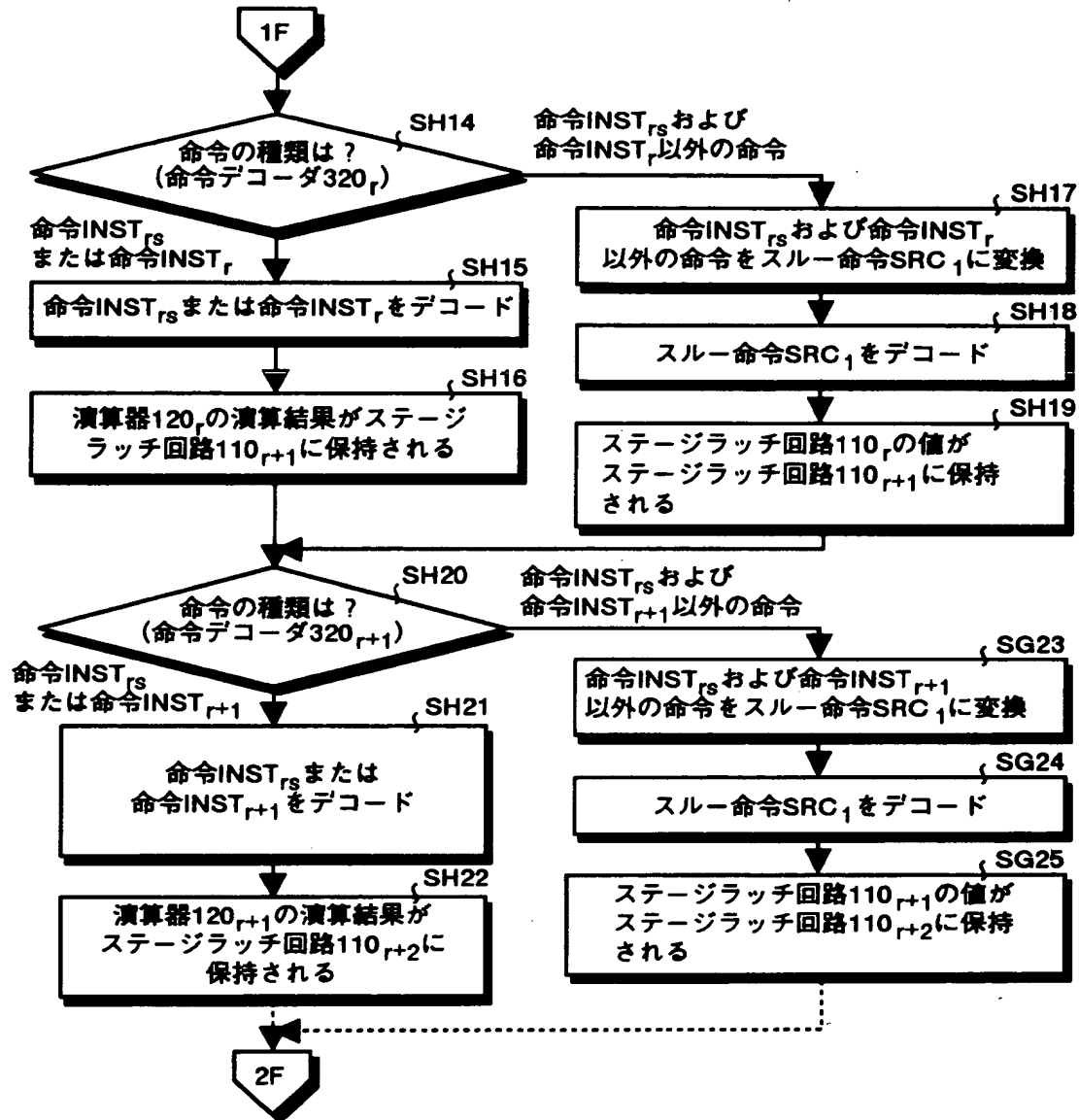


【図 2 4】

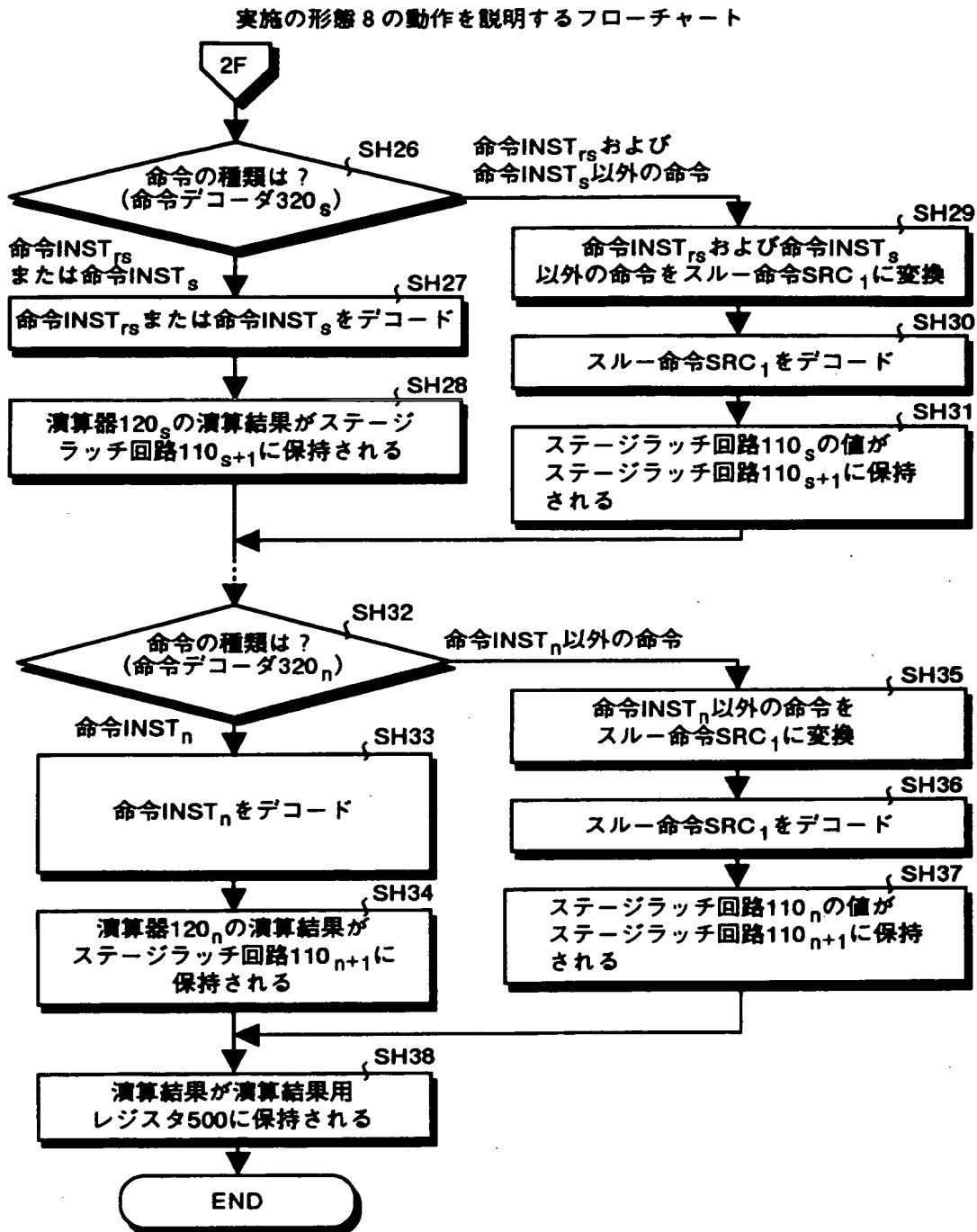


【図 2 5】

実施の形態 8 の動作を説明するフローチャート

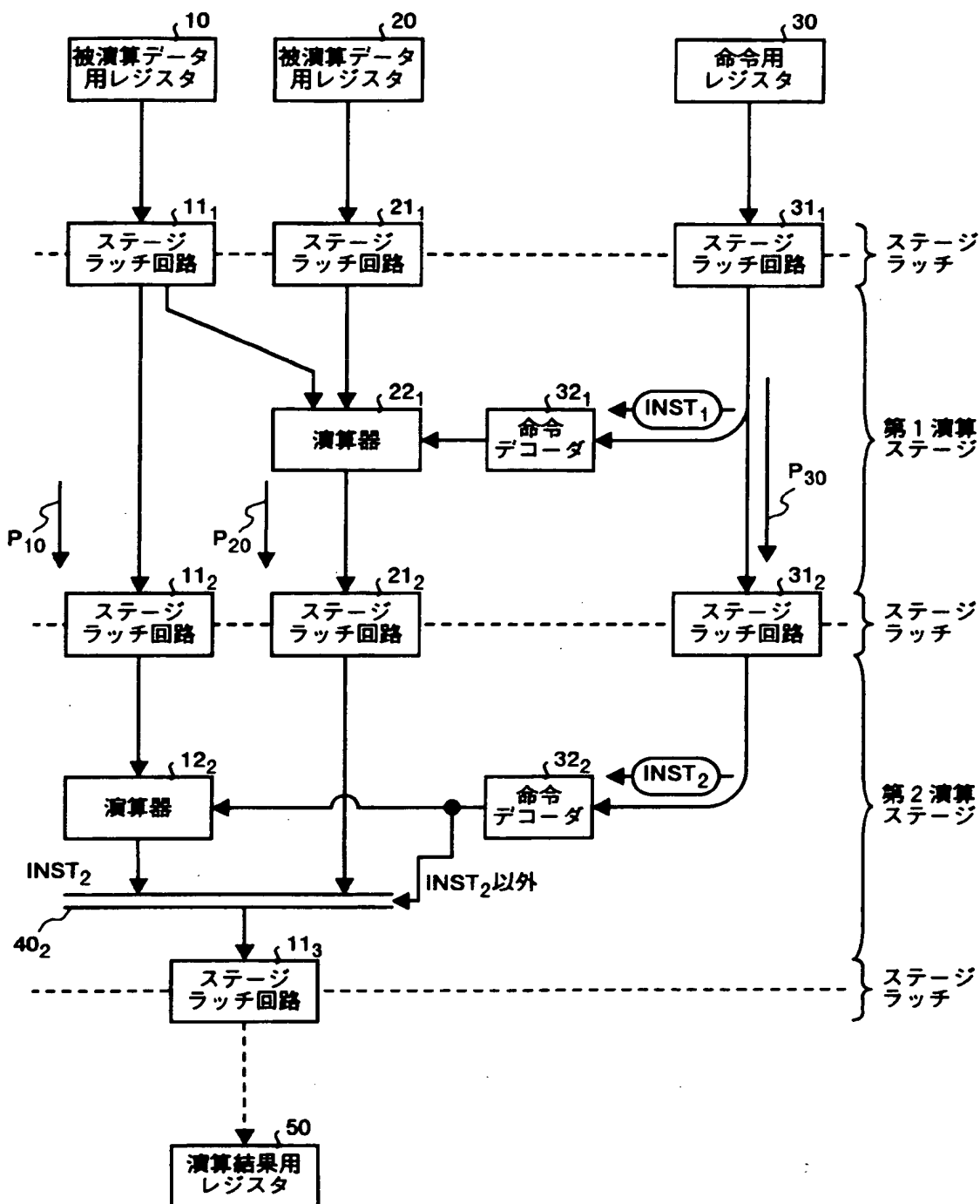


【図 2 6】



【図 2 7】

従来のパイプライン演算装置の構成を示す図



【書類名】 要約書

【要約】

【課題】 ハードウェア量、消費電力を低減すること。

【解決手段】 第 1 演算ステージの入力段に設けられ第 1 被演算データ SOURCE₁ および第 2 被演算データを保持するステージラッチ回路 1 1 0₁ およびステージラッチ回路 2 1 0₁ と、第 1 演算ステージに設けられ第 1 被演算データ SOURCE₁ および第 2 被演算データを用いて演算を行う演算器 1 2 0₁ と、第 1 演算ステージと第 2 演算ステージとの間に設けられ演算器 1 2 0₁ の出力値を保持するステージラッチ回路 1 1 0₂ と、第 2 演算ステージに設けられ命令 INST₂ がデコードされたときステージラッチ回路 1 1 0₂ の値を用いて演算を行う演算器 1 2 0₂ と、上記命令 INST₂ を、ステージラッチ回路 1 1 0₁ の値（第 1 被演算データ SOURCE₁）をスルーさせるスルー命令 SRC₁ として演算器 1 2 0₁ に対してデコードする命令デコーダ 3 2 0₁ とを備える。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号

[000005223]

- | | |
|----------|-----------------------|
| 1. 変更年月日 | 1996年 3月26日 |
| [変更理由] | 住所変更 |
| 住 所 | 神奈川県川崎市中原区上小田中4丁目1番1号 |
| 氏 名 | 富士通株式会社 |